AUTONET 2030

SEVENTH FRAMEWORK PROGRAMME

7th Framework Programme

Project number 610542

D1.3 Public Final Report

Work package	WP1	Project Management
Editor(s)	Panagiotis Pantazopoulos (ICCS)	
Status	Final	
Distribution	Public (PU)	
Issue date	22 November 2016	
European Commission Information Society and Media	Project co-funded by the European Commission DG-Information Society and Media in the 7th Framework Programme	SEVENTH FRAMEWORK PROGRAMME

Document information

Authors
Angelos Amditis – ICCS Panagiotis Pantazopoulos – ICCS Panagiotis Lytrivis – ICCS Andreas Festag – TUD/Fraunhofer IVI Xiangjun Qian – ARMINES Arnaud de La Fortelle – ARMINES Marcus Obst – Baselabs Pär Degerman – SCANIA Filippo Visitainer – CRF Luciano Altomare – CRF Guillaume Jornod – EPFL Alcherio Martinoli – EPFL Andras Kovacs – Broadbit Energy Technologies Laurens Hobert – HITACHI
Document editor
Panagiotis Pantazopoulos - ICCS Tel: +30 2107721076 Email: ppantaz@iccs.gr
Project funding
7th Framework Programme FP7-ICT-2013-10 Collaborative Project Grant Agreement No. 610542

Revision and history chart

Version	***Date***	***Comment***
0.1	18.10.2016	Initial document outline (ICCS), ToC
0.2	3.11.2016	Include inputs from partners (TUD, ICCS), include comments from CRF
0.3	14.11.2016	Include inputs from partners (ARMINES, Baselabs, Scania, CRF, EPFL)
0.4	15.11.2016	Include inputs from partners (Broadbit), add introduction/objectives, formatting
0.5	16.11.2016	Formatting, inputs to subsections 1.2, 2.2, and Sections 6 and 7, include inputs from partners (HIT, Broadbit lessons learned)
0.6	17.11.2016	Inputs to Sections 1.2, 2.1, 2.2, introductory paragraphs to various Sections, include HITACHI inputs to paragraph 7.1.4
0.7	18.11.2016	Formatting, proof-reading and improvements, ready for internal review
0.8	20.11.2016	Update of ARMINES references, integrate TUD, SCANIA, ICCS comments
0.9	22.11.2016	Final version

Table of Contents

Table of Figures

Executive Summary

This document serves as a publicly-available reference-document (typically referred to as the Final Report) that presents the overall work and achievements of the EU co-funded AutoNet2030 research project. The latter aimed to design, develop and validate a *cooperative* automated driving technology, based on a decentralized decision-making strategy and enabled by the *mutual information sharing* among nearby vehicles.

To realize the required functionality for vehicle-automation the AutoNet2030 approach included the careful system design, implementation, vehicle-integration and validation of a complex software system that was composed of four main parts: the maneuver controller, the perception layer, the communication stack and the human-machine interface (HMI). The corresponding challenges faced along each of these threads of research-work as well as the adopted approaches (by the consortium) and their evaluation, are herein presented in a brief yet systematic way.

For the sake of completeness this report also includes summaries of the ITS standardization work that the consortium has contributed to relevant International Organizations, the dissemination procedures and results as well as some directions of future exploitation of the AutoNet2030 achievements. It concludes with useful lessons and hints related to the actual realization of automation technologies as taken-out of the consortium's expertise and technical experiences during the last three years.

The achievements and results of AutoNet2030 are expected to find their way to real-world deployment during the 2020-2030 time horizons relying on the increasingly pervasive pillars of *cooperation* and *communications* that complement the matured sensor-based technologies. With its successful demonstration of cooperative automation under certain scenarios of both industrial and societal value, the AutoNet2030 project can claim considerable contributions towards shaping the path for cost-optimized and widely deployable automated driving technology.

1 The AutoNet2030 Project

1.1 Introduction

Vehicle-automation capabilities will hold a key-role in future mobility as they can provably provide safer driving conditions, improved comfort and more efficient traffic management. The so-far relevant research mainly aims at addressing different aspects of pure sensor-based vehicular technology such as sensing capabilities, V2X communications or vehicle control algorithms. Individual progress made along these threads has contributed to the deployment of integrated ADAS (Advanced Driver Assistance Systems) with increased levels of automated functionalities. However, highly automated vehicles are yet to come in mass-market deployment; the latter achievement calls for comprehensive investigation of the *complementarity* between the on-board sensory equipment, the wireless V2X communications and the distributed control algorithms.

Triggered by the so-far limited convergence between sensor-based automation and cooperative V2X communications, the AutoNet2030 project [**1**] has invested effort on research and validation of procedures and algorithms for 802.11p-based interactive control among co-operative (automated and manually-driven) vehicles focusing on:

- Cooperative decentralized control system to realize fully-automated vehicles and drive the advised maneuvering of manually-driven vehicles.
- V2X communications to enable automated maneuver planning and traffic flow optimization, which have been fed to ETSI ITS standardization groups.
- Onboard sensor-based architecture to enable reliable positioning and lane-keeping automation.

The document at-hand, first, presents a short introduction to the current and emerging ITS landscape as well as the AutoNet2030 concept. Then, it outlines the achievements of the AutoNet2030 project, both in terms of solutions to open technical challenges along the above research threads but also in terms of the project's outreach to the ITS community and the corresponding dissemination results. The involved target audience of this report as well as the AutoNet2030 work in-general, is comprised by a broad set of interested ITS stakeholders; it includes OEMs, tier-1 suppliers, SMEs and most notably the ITS research community, either academic or industrial.

The herein presented information constitutes only a part of the work and results achieved by the consortium throughout the project's lifetime and as a public document, seeks to attract the interest of the ITS stakeholders and increase the project's visibility. More technical details can be found at the project's website and the publicly-available deliverables:

- D3.2 – "Specifications for the enhancement to existing LDM and cooperative communication protocol standards"
- D5.2 – "Prototype validation report and performance analysis"
- D6.1 – "Report on the performed cooperative vehicle automation related standardisation activities and standardisation status"

Conditionally, such details can also be provided by the project partners, upon request.

1.2 The AutoNet2030 Vision

Through the past decades, innovation within the automotive sector has been incremental towards manufacturing safer and cleaner vehicles. More recently, the challenge has been to develop technology that will allow the transition from conventional driving to semi- and even fully-automated driving. Towards that end, the common vision among the AutoNet2030 consortium regarding the (near) future of automation and the involved challenges, amounts to a fleet of automated vehicles leveraging *communications* and *cooperation* among them (and also infrastructure) that will provide higher situational awareness.

The so-called *Connected Automated Driving (CAD)* paradigm (see Figure 1) is expected to drastically improve safety, comfort as well as traffic efficiency compared to pure sensor-based approaches to vehicle-automation. However, the turning of this vision into reality requires the *convergence* between stand-alone vehicle automation and cooperative V2X communications. The latter remains an open ITS-research issue where the AutoNet2030 project aimed to contribute by developing and also demonstrating of an instance of CAD technology.

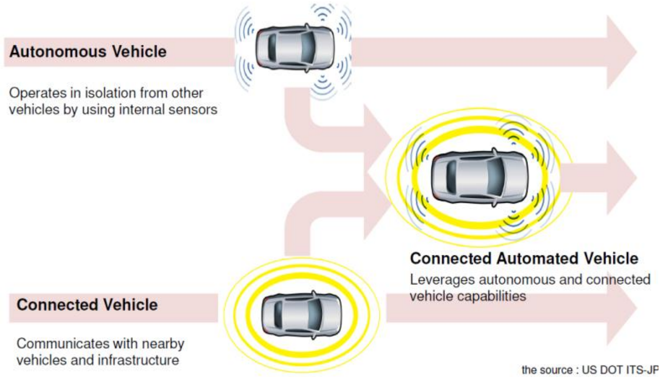


Figure 1 – Towards cooperative automated driving (*photo courtesy of US DoT ITS Joint Program Office*)

In line with the above emerging paradigm, a number of involved parameters/attributes pose further challenges in the task of deploying the relevant CAD technology. Keeping the complexity low is a key-point for the CAD vision as the latter involves the close coupling of multiple different technologies. Furthermore, what the AutoNet2030 project has identified in the emerging environment of CAD and tried to cope-with is summarized in the following points:

- The emerging fleet would include vehicles of various automation level, with different needs and level of equipment (*i.e.*, capabilities). In that sense, solutions of high-interoperability attributes are being sought-for.
- Appropriate information sharing is needed to drive the cooperative decision-making. In that sense, communications and exchanged messages hold a critical role. When a vehicle starts communicating it is also very important to consider privacy of the vehicle driver and owner. Another important fact is that viable business models of current and future services can be ensured by various stakeholders.

- Traffic efficiency issues need to be addressed in diverse settings (highway/urban) and different levels of mixed traffic. In that sense, any cooperative automation system needs to take into account the presence of manually-driven vehicles. Advised maneuvering is typically needed to cope with them.

1.3 Objectives

The project's objectives were essentially shaped by the AutoNet2030 vision of the emerging/future automation landscape and the corresponding challenges. Central to the vehicle automation environment of the near future would be on the one hand, the aspect of cooperation and on the other the presence of a mixed fleet of vehicles. Along these lines the project has made a distinction and explored automation use-cases along two different *interactions* among *cooperative* vehicles:

- Interactions among cooperative automated vehicles which appears to be the most predictable one; it involves the exchange of planned trajectories and negotiation of maneuvers.
- Interaction between a cooperative automated vehicle and a cooperative manually driven vehicle: this interaction involves the exchange of vehicle's state variables (*e.g.*, position, speed, acceleration) and the planned route. Towards that end, appropriate V2X messages that fulfil certain requirements need to be designed and introduced (see section 4.3.2)

Accounting already for the above interactions, the project identified and tried to achieve a number of high-level goals:

- To achieve increased awareness and higher confidence perception layer relying on communications,
- To explore the transition from co-operative perception (*i.e.*, exchange of position & sensor data) to maneuvering negotiation,
- To design and develop vehicle-automation technologies keeping the system complexity as low as possible,
- To provide experimental evidence that cooperative automation can be efficiently realized in highway/urban settings under *mixed* vehicles fleet, and
- To contribute to relevant standardization, validation, certification processes

When moving to the actual modules and the corresponding system functionality the high-level goals can be mapped into certain objectives (pointing to the relevant design and implementation tasks):

1. Cooperative ITS communication system including new protocols
2. Cooperative perception
3. Cooperative decision making (*i.e.*, starting of maneuvers)
4. Cooperative motion planning (*i.e.*, ensuring that motions are feasible, safe and well coordinated)
5. Cooperative control (*i.e.*, each vehicle performs its control and ensures it is coordinated)
6. HMI supporting cooperation for manually driven cooperative vehicles

The objectives 3-5 can be commonly called "cooperative maneuvering"

2 Concept and Considered Use Cases

This section elaborates on the automated-driving concept and the way that the AutoNet2030 cooperative system seeks to extend the (typical in automated-driving) exchanging of positioning and sensory data to maneuvering negotiation under a number of carefully-selected use-cases.

Related deliverable:

D2.1 Cooperative automated driving use cases and requirements

2.1 The Automated-driving Concept: State-of-the-Art Approaches

The AutoNet2030 project focuses on networked cooperative vehicle automation. The central point in the AutoNet2030 concept is the *cooperation* between networked vehicles and the way to maximize the benefits out of it. At the same time the involved fleet of vehicles would include both automated and manually-driven ones calling for both advanced automation features (of SAE level 3) and advised maneuvering, respectively. The work carried-out in the context of the project used previous well-known results and built upon the latest advances achieved by relevant research.

The European Union has a long history of investing funds and effort on collaborative research projects contributing to the development of automated driving. During the last ten years or so, a considerable number of R&D projects have managed to produce significant results in vehicle automation that shape the state-of-the-art (SoA). A brief overview of these projects and their outcome is presented below (in a non-exhaustive list):

- The R&D Integrated Project (IP) PreVENT brought together a large consortium from automotive industry and research to develop, test and promote the Advanced Driver Assistance Systems (ADAS). The project successfully combined sensor information with communication and positioning services to improve the driving safety.
- The R&D project HAVEit put-together a network of 17 European partners from automotive industry and universities aiming to increase driving safety and boost the European automotive industry in an international market. A number of automation modes such as lane keeping assistance or an emergency braking assistance were designed and implemented.
- The R&D SARTRE Project targeted towards exploring and promoting road-trains (platoons). Platooning applications were tested for high-speed driving on public motorways. The COMPANION project (funded by the same EU call as AutoNet2030) appeared as a SARTRE successor exploring the practical use of platooning in transport operated by heavy-duty vehicles.
- The R&D project InteractIVe gathered 29 companies from 10 countries to work for the increase of European accident-free traffic. Towards that end the consortium designed and developed ADAS systems for safer and more efficient driving.

- An integration of driverless intelligent vehicles (of SAE level 5) in urban environments was made by the R&D project CityMobil2. The project implemented systems for automated transport in several protected environments around Europe. Relevant scenarios involve the transport of passengers (over loop lines connecting locations of interest) by electric vehicles moving into separate road areas, excluding risks caused by the presence of other vehicles.
- Finally, the on-going European project AdaptIVe stands as the flagship integrated project in Europe. Its large consortium of 29 partners aims at the development of a broad set of features for automation under different driving scenarios. The achieved improvements are expected to lead to safer and more efficient automated driving.

These successful project activities (together with their industrial counterparts) have contributed to the advance of driver assistance systems in recent years. The advanced driver assistance systems (ADAS) such as adaptive cruise control (ACC) and lane departure warning (LDW) or lane keeping assist systems (LKA) appear today as mature technologies. On the other hand, current automated vehicles can only perform self-drive under limited conditions: significant technical challenges must be addressed. One of them, not adequately addressed by the aforementioned projects, is the way to realize cooperation by implementing the functionality for maneuvering negotiations. There, the AutoNet2030 project has showcased promising results that can assist to realize the connected automated driving on the roads sometime during the 2030s.

A final note on the way to advance SoA vehicle automation relates to the importance of technology demonstration. Self-driving solutions, even if far from mature, when presented to public can work as a pull for innovation for the research community and at the same time raise awareness of the potential benefits of automated driving among stakeholders and future customers. Thus, a broader audience is enabled to get familiar with the new technology.

The core of the AutoNet2030 demonstration *i.e.*, the selected use cases are summarized next.

2.2 Considered Vehicle-automation Settings

AutoNet2030 has selected to focus on two vehicle-automation settings of diverse characteristics. In this way the project had the opportunity to test, validate and demonstrate the performance of the prototype system over the broadest-possible set of (future) driving instances. In each setting a number of use-cases (*i.e.*, presentation scenarios) were demonstrated showcasing a different subset of the project's objectives (see Section 1.3). The use-cases of the two settings even though completely different, used similar algorithms whenever applicable.

2.2.1 Highway Setting

The motivation of selecting the highway setting (*i.e.*, speed up to 70-80km/h) relates to the convoy motion that is expected to hold a central role in upcoming instances of safe and cost-efficient mobility. The formation of convoy motion together with the capability of surrounding vehicles, whether manually-driven or automated, to interact with it, is expected to become a necessity in the near future.

The use-case performed in highway-like conditions (at AstaZero proving ground) used one heavy-duty truck (Scania), one automated passenger car (Hitachi vehicle) and one completely manually-driven passenger car (Fiat 500L). The presentation story is actually divided into three distinct scenarios, all showcasing different AutoNet2030 use cases where cooperative functions is the overall theme.

In the first use-case, the truck is driving along a highway-like road at approximately v_{ego} = 60 km/h. The automated car catches up in the same lane. When the car approaches the truck (*i.e.*, in communication distance) they join and form a convoy motion at v_{ego} = 70 km/h. In this case the cooperative ITS communication system, the cooperative control and HMI features are demonstrated.

In the second use-case, the two cooperative automated vehicles (from the first use-case) above drive in convoy along the highway-like road. A manually-driven vehicle approaches and intents to merge into the same lane as the two convoy vehicles. The convoy vehicles increase their in-between gap and the car safely joins the convoy relying on HMI projections. All three vehicles are driving in cooperative mode, in the same lane at v_{set} = 70 km/h. The manually-driven car uses HMI projections to advise the driver on how to drive safely in the cooperative cluster staying for some time in the lane and keeping a constant speed. By using employing *cooperative decision-making* and *maneuvering*, the merging was performed in a safe and efficient way.

Then, the manually-driven vehicle decides to leave the cooperative cluster, and does that by changing lane and pulling away from the two vehicles. The latter ones recognize the leaving and close the distance. In this case the cooperative ITS communication system, the cooperative control and HMI features are demonstrated.

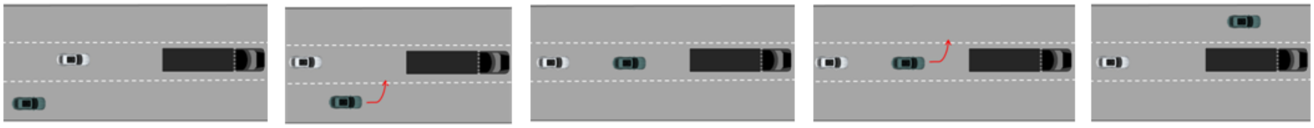


Figure 2 – An instance of the AutoNet2030 use-cases performed in highway setting

It is important to note that a third more demanding use-case was successfully demonstrated by the project. It evolved under the same scenario as above (see above figure) up to the merging of the car into the convoy. The difference compared to what described above was the fact that the two passenger cars had exchanged roles; the car performing the merging maneuver (*i.e.*, dark car in the above figure) -in this case- was automated. The impressive result was one of the AutoNet2030 highlights.

2.2.2 Urban Setting

The other demonstration scenarios take place in a more inner city-like setting, where fully automated, electric prototype vehicles showcase the AutoNet2030 results under low-speed driving (*i.e.*, lower than 25km/h). In urban settings there is already a need to efficiently coordinate automated electric vehicles (typically used for short-range transport).

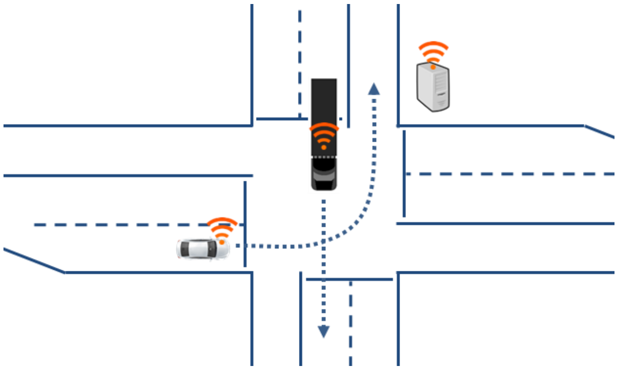


Figure 3 – An instance of cooperative intersection control

The relevant demonstration was performed and filmed at the INRIA premises (France) and was later showed as a video presentation at the AutoNet2030 final event.

- Car following. Two vehicles are driving in a lane at low speed, maintaining a tight formation (*e.g.*, constant time gap). The follower can also immediately respond when the leading vehicle performs an emergency brake. The use-case demonstrates how the vehicles can keep a perpetually safe relation in car following scenarios relying on cooperative communication and control.
- Intersection crossing (*i.e.*, merging of vehicles into the same road). Two vehicles are coordinated to merge into a main road. The merging orders are decided in an automated way. The vehicle having no priority decelerates to provide sufficient merging space. The merging operation is successfully performed (*i.e.*, no collision occurs) and then the relation of two vehicles becomes normal car-following. Using *cooperative maneuvering*, the merging can be performed in a safe and efficient way. The reliability and benefits of Cooperative communication and cooperative sensing are also demonstrated.

The design and development of the different AutoNet2030 modules needed to realize the use-cases in both settings, is detailed in Section 4.

2.3 Supporting Sustainable Safety

Road accidents remain among the leading mortality causes and highlight the need for more efficient deployment of transportation safety solutions. An apparent tool to fulfil the EU's ambition in order to decrease the number of road fatalities by 50% between 2010 and 2020 is vehicle automation. Indeed, the generally anticipated path to fatality-free road transportation relies heavily on fully automated vehicles, which are impossible to crash under normal operation.

Initial vehicle automation research has considered handling the extreme scenarios of uneventful highway driving on the one hand and dangerous traffic incidents on the other hand. From these initial explorations, further research has extended the scope of scenarios to more complex ones reaching settings of collective motion. The AutoNet2030 in particular has explored highway convoy-driving scenarios as well as intersection coordination in urban settings.

Employing precise longitudinal and lateral vehicle control, the maneuvering control algorithms such as the ones developed in AutoNet2030 are enabled to facilitate fatality-free road traffic precluding any dangerous incidents. The relevant guarantees stem from the fact that the proposed control algorithms are based on mathematical models with provable performance characteristics, and since these control components receive input through redundant perception sensors, it can be reasonably expected that the project's testing results imply equivalent performance upon real-world deployment. Care must be taken though, to ensure that the used implementations scale well to a higher number of vehicles. Within the boundaries of the project, the algorithms have already been proven in simulation.

On top of that, AutoNet2030 has strongly promoted, implemented and validated in *several realistic test-track sessions with prototypes* the dimension of cooperation in vehicle automation. Contrary to pure sensor-based technologies the cooperative automation functionalities demonstrated by the project present a strong potential to impact the vehicle safety. Along this line, one of the project's achievements has been the proposed extension to the CAM messages so as they include trajectory-related information (see Section 4.3.2). The analysis of the relevant field-testing results together with the relevant simulation studies can bring-about further improvements to future transportation safety.

3 The AutoNet2030 System and Software Design

This section discusses some of the design principles of the AutoNet2030 system [**1**]. First, we consider the way it is structured to cope with the general automotive setting and then focus on the software modules that comprise the AutoNet2030 vehicle platform.

Related deliverables

D3.1 Initial system design specifications for automated driving support (*This deliverable presents a preliminary definition of the functional architecture of the system responding to the earlier-identified requirements*.)

D3.3 Final system design specifications for automated driving support (*This deliverable presents the final definition of the functional architecture of the system responding to the earlier-identified requirements*.)

3.1 The AutoNet2030 System Specifications

In line with the research work carried out in WP2 and the earlier presented use-cases (see Section 2), a list of automated driving control use cases and requirements was compiled. This section describes the specifications of AutoNet2030 system that can execute the use cases and satisfy the requirements.

In AutoNet2030, we have conceived a complete, self-contained system for cooperative automated driving in both highway and urban scenarios. The system is composed of three subsystems as shown in Figure 4, briefly introduced in the following:

- **Onboard Subsystems**: Onboard subsystems are integrated in cooperative vehicles and consist of the hardware and software needed to support the AutoNet2030 use cases. The hardware includes onboard sensors to sense the environment (e.g. (stereo) camera’s, radar’s, etc.), actuators, an HMI, communication technology (e.g. IEEE802.11p, LTE) and processing technology to run the AutoNet2030 control, perception and communication algorithms.
- **Roadside Subsystems**: Roadside subsystems are optionally installed alongside the road in order to support onboard subsystems in maneuver negotiation and provision of information from the infrastructure. For example, the roadside subsystem may be installed at an intersection in order to support the maneuver negotiation between vehicles trying to cross the intersection. The roadside subsystem is equipped with the appropriate communication (e.g. IEEE802.11p, LTE) to interact with other subsystems of the AutoNet2030 system.
- **Central Subsystem**: the central subsystem is a system that runs in the cloud and is accessible over the internet by on-board subsystems and roadside subsystems. The central subsystem has two main functionalities. Primarily, it is supporting the communication in the global cooperative area (GCA) between e.g. on-board subsystems. Secondly, it is distributing information to roadside subsystems such as RTCM data. The AutoNet2030 system allows for multiple application-specific central subsystems which optionally inter-exchange information.

Figure 4 - AutoNet2030 system structure

We have analysed and designed the architecture for each subsystem, with the focus on the on-board subsystem. Figure 5 is the block diagram illustrating the hierarchical and modular design of the on-board subsystem.

From a modular viewpoint, the subsystem is divided into four major modules: maneuver controller, perception, HMI and communication. Each module can be further divided into components. The interaction between modules is achieved by pre-defined interfaces. The maneuver controller is the central hub of vehicle maneuver control. If the vehicle is in cooperative automated mode, the maneuver controller calculates feasible trajectory and generates maneuver commands according to the information provided by communication and perception components. If the vehicle is in cooperative manually-driven mode, the maneuver controller generates advices to the human driver and demands the HMI system to display the generated advice. The perception module comprises interfaces to physical sensor systems as well as algorithmic parts that refine and derive a comprehensive representation of the complex environment around the host vehicle which is needed for cooperative autonomous driving. Furthermore, this perception information is stored in a unified format and provided as a service to other components of the AutoNet2030 on-board subsystem. The HMI module is responsible for driver’s notifications and interactions in all driving scenarios. The communication module supports Perception and Maneuver Control components for diverse functionalities that requires V2V/V2X communications, such as cooperative convoy control and cooperative intersection management.

In a hierarchical viewpoint, the subsystem is divided into four layers inspired from Open System Interconnection (OSI) model. The maneuver controller runs at the application layer to provide maneuver guidance and vehicle control for cooperative automated driving. Perception, HMI and a part of communication runs at the facility layer to provide sensing, human machine interface and vehicle to vehicle/infrastructure communication services to the maneuver controller. Finally, various communication protocols are used at C-ITS networking & transportation layer (NWT Layer) and access layer to support communication services.

In this document, the overview of the specification is provided. Detailed system specification is available in deliverable D3.3 final system design specifications for automated driving support. Note that the AutoNet2030 methodology is to provide a comprehensive view on the system specification, thus some parts of the system are specified in detail while are possibly not actually implemented due to the constraints on resource, time and interests of partners.

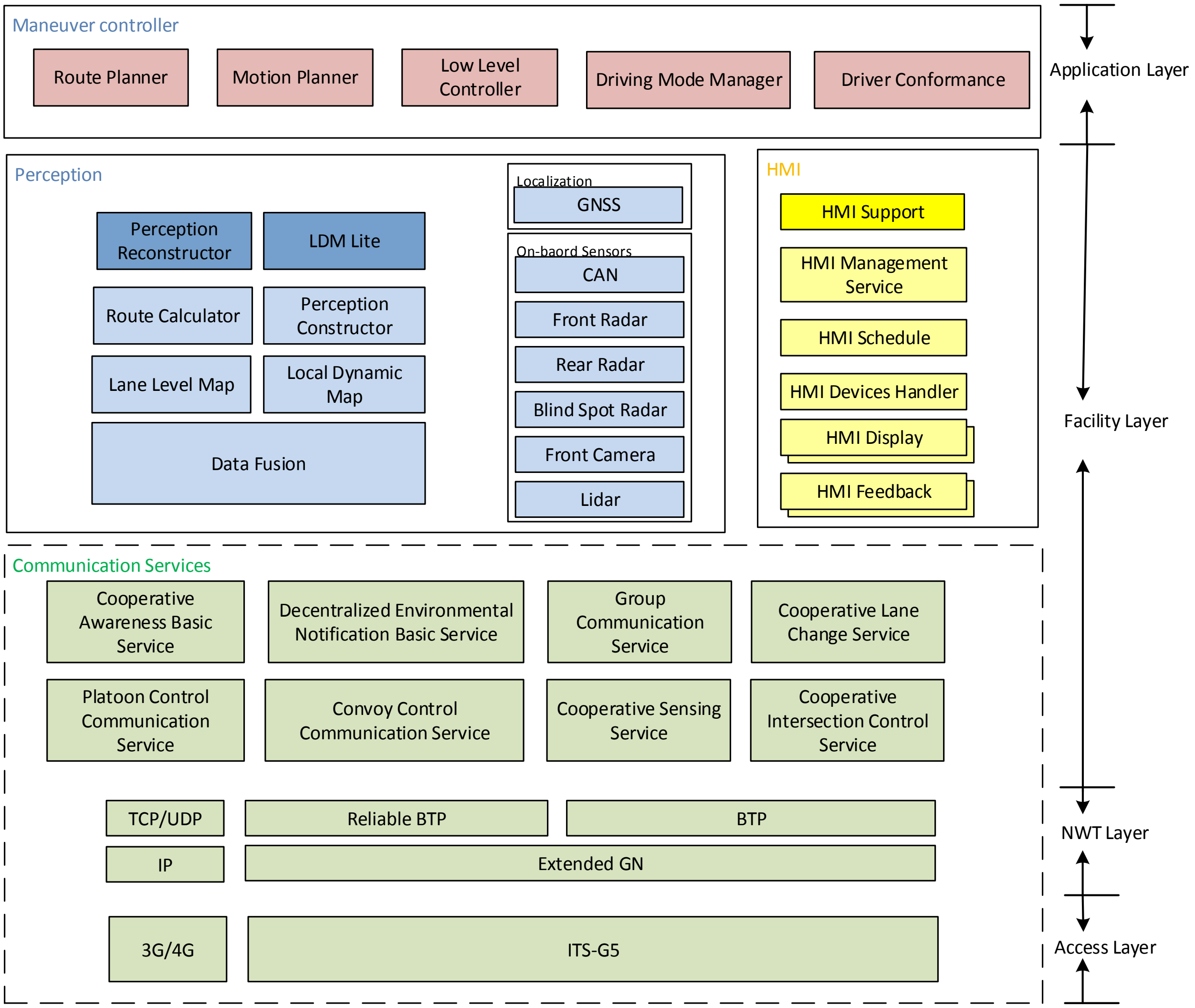


Figure 5 - Component diagram of AutoNet2030 vehicle software architecture (on-board subsystem)

3.2 The AutoNet2030 Software Modules and Architecture

The AutoNet2030 on-board subsystem comprises four main modules: Communication Services, Perception, HMI and Maneuver Controller. In the following section these components and their interaction are briefly introduced. We again refer to Figure 5 that illustrates the whole architecture from the access layer towards the application.

This Maneuver Controller is the central hub of vehicle maneuver control. If the vehicle is in cooperative automated mode, the Maneuver Controller calculates the feasible trajectory and generates maneuver commands according to the information provided by the communication and perception components. If the vehicle is in cooperative manually-driven mode, the Maneuver Controller generates advices to the human driver and demands the HMI system to display the generated advice.

The perception was installed on any AutoNet2030 vehicle in order to provide a unified view of the environment around the vehicle. Besides information about the ego vehicle such as global position, velocity and acceleration, the perception models the dynamic and static environment around the host vehicle. The perception abstracts particular sensor interfaces and vehicle characteristics and thus provides a sensor-agnostic service towards the facility and application layer. More specifically, it includes a multiple object tracking, free-space estimation and a GNSS/INS localization system.

The communication services support the perception and maneuver control components for diverse functionalities. This includes the provision of the contents from received CAMs as well as AutoNet2030 specific V2V message introduced for cooperative coordination. Furthermore, this module supports the cooperative sensing service (CSS) to distribute accumulated sensor data via ITS-G5.

The HMI system of a vehicle, in general, is the OEM-system component responsible for driver's notifications and interactions. In the AutoNet2030 project, the HMI system is either an internal or external part of the onboard subsystem. This decision is made by the provider of the prototype. The HMI system operates with the help of the HMI Support facility that decouples application logic from any OEM-specific HMI properties.

The integration of the four main components within the AutoNet2030 vehicles was achieved via several techniques. For the perception and the maneuver controller, isolated and self-contained application containers (based on the Docker infrastructure [**2**]) were prepared and deployed to the central computing platform within each vehicle. By that, easy updating as well as version-management and rollback functionality was ensured. The interfaces between these containers and the remaining components (HMI, communication services and sensor interfaces) were realized via LCM [**3**], a lightweight UDP-based broadcast protocol that also supports efficient serialization and deserialization of various sensor and control data. The HMI application itself was implemented on an Android OS-based application and connected to the other components via a JSON-based network interface. In the VW Passat, the HMI was directly displayed within the integrated infotainment unit of the vehicle. For the other AutoNet2030 demo vehicles a separate and wireless connected tablet was used. Most of the communication services were directly implemented on a dedicated ITS-G5 device. This device was also connected via LCM to the other parts of the AutoNet2030 system.

4 The AutoNet2030 Achievements

This Section presents a summary of the main technical work undertaken in the context of the AutoNet2030 project. Along the four main parts of the AutoNet2030 system *i.e.*, the maneuver controller, the perception, the communications stack and the HMI, all corresponding work is highlighted and the way to address the involved challenges is briefly explained.

Related deliverables:

D4.1 Component development report (*This deliverable reports on the development progress of the novel control algorithms, the digital maps (LDM) as well as the communication protocol implementations*)

D4.2 Prototype development and integration report (*This deliverable describes the development and the integration of the AutoNet2030 prototype including also implementation details for the HMI system)*

D5.1 Test plan (*This deliverable contains the project's test plan that highlights experimentation activities conducted both in the laboratory environment but also in the test-track proving ground*)

D5.2 Prototype validation report and performance analysis (*This deliverable contains the results of the validation experimentation of the developed modules and the integrated AutoNet2030 system*)

4.1 Maneuver Controller

The AutoNet2030 work related to the control and maneuvering of automated vehicles[1] has delivered two relevant instances exploring systematically the relevant solution space. The first amounts to a distributed graph-based controller tailored for the convoy motion of vehicles. The other is a semi-distributed hierarchical control framework that can also support the convoy motion but for the need of demonstration is applied on intersection management (where priorities are carefully assigned to automated vehicles). Those two instances have been developed to address in a *fail-safe manner* the peculiarities of both considered AutoNet2030 vehicle-automation settings; the first instance is for high-speed driving (*i.e.*, highway scenarios) while the second is for low-speed (*i.e.*, urban scenarios).

4.1.1 Distributed Convoy Controller

To address the problem of convoy formation, maintenance and maneuvering, a distributed convoy controller has been designed, tested, implemented in simulation and integrated within the AutoNet2030 system on the different vehicles.

The approach of convoy formation in this context is driven by the absence of a single leader of the convoy, which would be followed by the other members of the convoy. The leaderless convoy is also not ruled by a

---

[1] Directives for the driver of a manually-driven vehicle can be also provided.

centralized controller, for instance a road side unit communicating the actuation commands to the neighbouring vehicles. Subsequently, each vehicle is running the same set of rules, leading to the formation of a leaderless, distributed convoy.

The core feature of convoy formation is consensus. In such a network of connected agents, it means to make all agents agree on their states, viz. their positions. A common solution to this problem is the Laplacian feedback control [**4**]:

$$\acute{x}(t) = -Lx(t)$$

where L is the Laplacian matrix. To achieve formation, this equation is biased using a bias matrix B [**4**], as illustrated by Figure 6.

$$\acute{x}(t) = -L(x(t) - B).$$

The biased Laplacian feedback has been applied to convoy formation with no road constraints in [**5**]. In this work, the transformation from velocity to control commands is done through the creation of a goal point (target speed) and a goal line (target line), assuming that the current speed and direction are known. As a result, two control laws lead the vehicle to reach these goal line and point.

The key input of the Laplacian graph-based control is positioning. To apply the Laplacian biased equation, an agent needs both its position and the ones from its neighbours. Laplacian graph-based control can be used with absolute positioning or with relative positioning. The former is generally obtained from raw GNSS, or GNSS

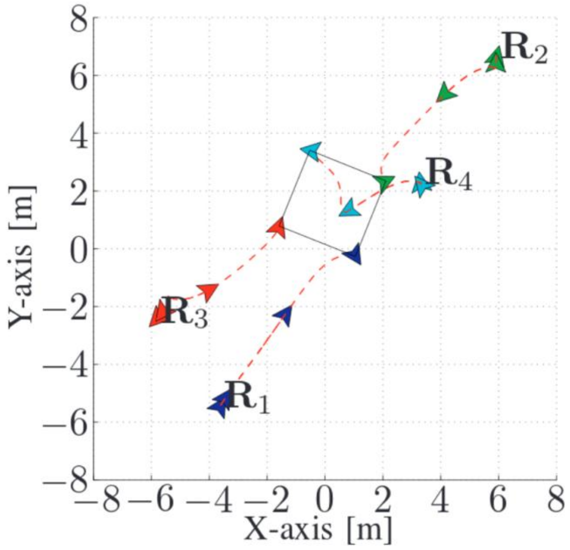


Figure 6 Evolution of a four-differential-wheeled-robot system converging to a square formation [6]

fused with inertial navigation and exteroceptive sensors, coupled with V2X communication for the neighbour inputs. The latter is usually given by ad hoc systems, or exteroceptive sensors such as cameras and LIDAR which data are processed by object recognition algorithms.

To adapt this work to the formation of distributed convoy on highways, the following assumptions have been made: (i) cars account for a close neighbourhood, (ii) they have no global knowledge of the total number of cars, the global shape, or the maneuvers occurring. Using local communication allows for flexibility in the convoy while ensuring a globally connected graph, resulting from the connection of the local graphs [**7**].

Knowing the position of the neighbouring cars, each car can maintain its local Laplacian and bias matrices. Figure 7 illustrates this matrix maintenance, showing the indexing of the neighbouring vehicles. This dynamic numbering allows cars to enter and leave the convoy, which is a first novelty reported in [**7**].

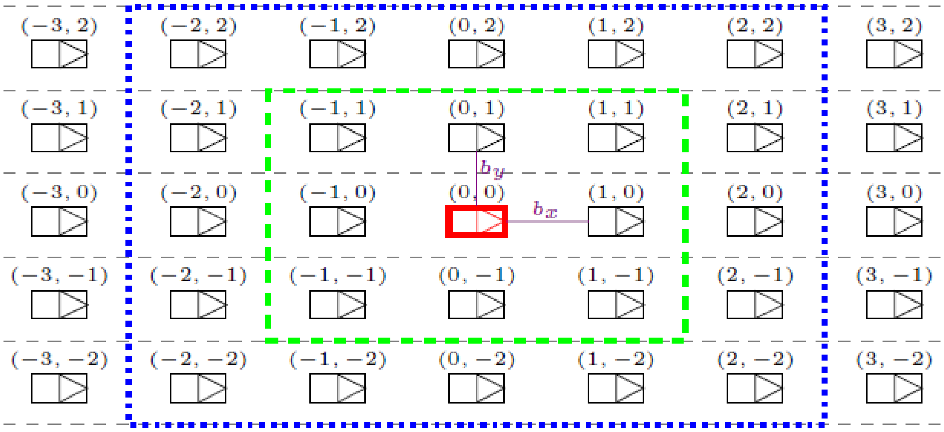


Figure 7 Indexing of the neighbours with different communication range from the point of view of the red car. [7]

The second novelty of this work [**7**] is the feature allowing the convoy to drive on the roads and to adapt its formation shape to the curvature of the lanes. To achieve this goal, the car requires the orientation of the road and it position with respect to the target trajectory. This information is either obtained through on-board exteroceptive sensors or from static maps combined with GNSS positioning. The former is a more dynamic and convenient when envisaging a long term application to any roads but add some computational costs. One would prefer the latter when considering multivehicle simulations.

As a result, we needed to choose between absolute and relative positioning and between exteroceptive sensors and static maps for road positioning. The perception module offers absolute ego-positioning, relative object positioning and both relative and absolute map data. For the sake of consistency and to take advantage of the available V2X solution, absolute positioning for both Laplacian control and map positioning has been chosen. Vehicles get their absolute positions from the perception module, localize themselves on a static map and broadcast this set of information to their neighbours.

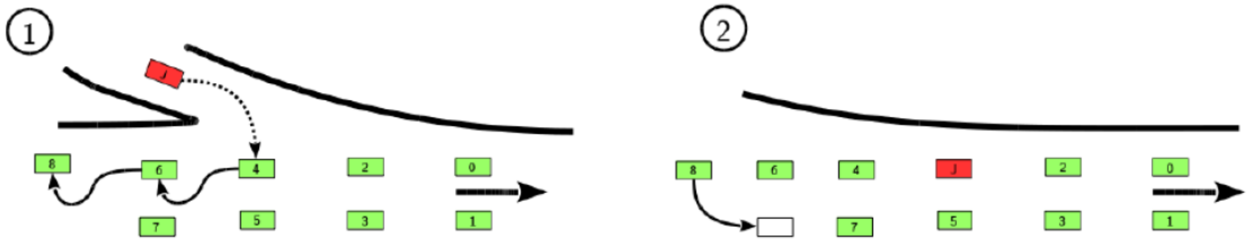


Figure 8 Maneuvers within the convoy: (1) Joining a convoy and (2) reorganizing the configuration [3]

In this specific first convoy control algorithm [**7**], multiple control laws (or behaviours) are concurrently operating on the actuators and an appropriate arbitration scheme needed to be proposed. In particular, the influence of the lane keeping, obstacle avoidance, and convoy formation control laws must be properly balanced. As a first attempt to solve this arbitration problem, we selected a simple and well-established behavioural arbitration scheme often used in mobile robotics [**8**] leveraging simply a weighted sum of the multiple concurrent behavioural outputs. This in turn required the introduction of additional control parameters (weights) and their subsequent optimization. We addressed this optimization problem using a well-established meta-heuristic method (Particle Swarm Optimization [**9**]) and generated multiple sets of parameters depending on the specific traffic scenario on which the control scheme above was deployed.

To fit with the use cases of AutoNet2030, in addition to the roads and dynamic number features, the algorithm also needs to be able to allow joining, leaving and lane changing operations. The local modifications of the graph can be operated thanks to the distributed and local characteristics of the algorithm. Propagated with multi-hop communication, the maneuvers are implemented for two lanes convoy, as illustrated by Figure 8.

All these features have been validated through test cases in simulation using the high fidelity simulator Webots [**10**] appropriately modified in parallel research and development efforts [**11**] to meet a sufficient degree of realism for automotive-relevant scenarios [**12**] .

The algorithm reported in [**7**] meets most use cases requirements, except that the computation of the bias matrix makes its shape rather rigid (as well as fixed to a rectangular coordinate frame) and does not allow different vehicle lengths. Moreover, the algorithm limits the dynamic operation to two lanes and is affected by delays and package losses.

We substantially modified the original algorithm to address the limitations above, resulting in a new solution recently published in [**13**]. A first key feature of the new algorithm is a higher sophistication in the level of cooperation among vehicles through the use of dedicated convoy messages. The convoy messages comprise content regarding the pose and velocity of the vehicle but also information about its length, current lane, and global offset within the convoy. Such messages are broadcast by each vehicle and used to construct the graph when received.

In particular, the broadcast length and offset of the vehicle is used to cope with heterogeneous vehicle lengths. With this information, a vehicle can create its local bias matrix in a computationally light way. The offset to the foremost vehicle also reduces the influence of the delay and package loss on the algorithm performance.

A second core feature of the new algorithm is the use of a curvilinear coordinate frame that relaxes the competition between the convoy-formation controller and the lane-keeping controller. Using the lane numbering (continuously broadcast through convoy messages) and the curvilinear longitudinal axis, the algorithm allows for formation of convoys and maneuvers in dynamically changing lane numbers.

The algorithm in [**13**] uses a finite state machine mechanism for enabling lane changing maneuvers. In particular, the vehicle states is augmented to include the different roles a vehicle can assume in such maneuvers (e.g., the vehicle actually intending to change lane or helper vehicles adjusting their position within the convoy to allow such maneuver).

The performance of this algorithm has been validated in simulation through a set of use cases using longitudinal, lateral errors and mean speed of the convoy [**13**].

This algorithm is also the one that has been integrated within the AutoNet2030 system and deployed on the real platforms. This integration has been validated through a set of test cases but also demonstrated at the final event at the AstaZero Facilities in October 2016.

Figure 9 AstaZero final event: FIAT 500L leaving the convoy

At the center of the architecture, the resulting controller has interfaces with almost all other AutoNet2030 components: the perception module for ego positioning and surrounding environmental information, including neighbouring vehicle data; the V2X for the broadcasting of convoy messages but also the control part of the CAM extension; the HMI for the display of convoy-related information, such as target speed, target lane, and convoy status; and the low-level controller to provide actual speed commands to the vehicles.

In this integration work, the high-fidelity simulator Webots [**10**] has been used as an integration test tool, where the code used in the real vehicle is interfaced with the sensors and actuators of the vehicle model in Webots. Other components, such as the perception module, also run in simulation while the low-level controller can only be accessed on the real platform, therefore being replaced by emulated low-level controller in simulation.

The use cases of the highway scenario demonstrated at the AstaZero facilities are the direct result of the successful deployment of the developed algorithm on the different platforms: one automated car, one automated truck and one manually driven car equipped with an appropriate HMI interface (see Figure 20). To cope with the different dynamic capabilities of the involved vehicles, a set of test in simulation allowed the choice of the right acceleration limitations to adapt the natively designed kinematic algorithm to the dynamically heterogeneous real convoy.

4.1.2 Hierarchical Control Architecture for Low-speed Driving

The focus of partner ARMINES is on the hierarchical approach. A hierarchical control architecture is proposed as illustrated in Figure 10. At high level, a centralized cooperative supervisor coordinates autonomous vehicles using a minimal set of rules. At low level, each vehicle has its own controller that performs trajectory planning and tracking control. Such design allows balancing the cooperative goals (intersection crossing, formation keeping, etc.) with individual interests (collision avoidance, comfort, etc.). Note that the supervisor (i.e. the centralized part of the algorithm) could be achieved using distributed techniques, thus it may be infrastructure-free, but there is a need of a global common view to synchronize the cluster of vehicles.

The vehicle controller can be further decomposed into two components: a motion planner in charge of trajectory generation and a tracking controller for vehicle control. The tracking controller was provided by car manufacturers and thus was not subject to AutoNet2030 research. We have proposed and tested two different designs of motion planner, all of them compatible with cooperative automated driving. The first design [**14**] uses Bézier curves to draw obstacle-free paths for the autonomous vehicle and then use Model Predictive Control (MPC) to parameterize paths with speed profiles. The advantage of this approach is its efficiency; the planning can be achieved in less than 50 ms. The second design [**15**] directly uses MPC to generate optimal trajectories for autonomous vehicles.

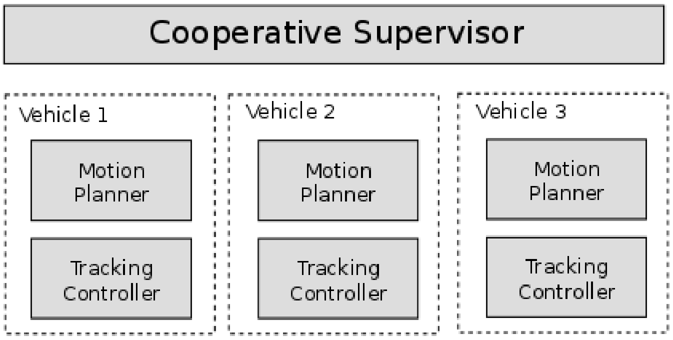


Figure 10 Semi-centralized cooperative automated driving architecture

We mainly consider two cooperative automated driving scenarios: autonomous intersection management and convoy. The cooperative supervisor is designed differently for two scenarios.

The AutoNet2030 system has provided unprecedented capacities to vehicles to precisely control the vehicle trajectory, percept surrounding environments and communicate with other vehicles and infrastructures. We fully exploit these capacities and propose a priority-based coordination framework [**16**] for autonomous intersection management. The cooperative supervisor assigns priorities to vehicles in an efficient way. Vehicles retain full control and plan trajectories that respect the assigned priorities.

For convoy control, the cooperative supervisor [**17**] maintains the adjacent matrices that describe the vehicle dependence relations and offset matrices that describe the relative distances between vehicles. This information is made available to individual vehicles so that they can plan locally their trajectories that maintain the shape of the convoy.

Technical details and in-depth analysis are available in research papers [**14**], [**15**], [**16**], [**17**].

The proposed semi-centralized control architecture has been validated in both simulations and experiments.

An example of simulations is illustrated in Figure 11, where a convoy of three vehicles proceeds in a highly constrained environment with obstacles. We note that the convoy is able to temporarily deform from the desired formation to avoid collisions.

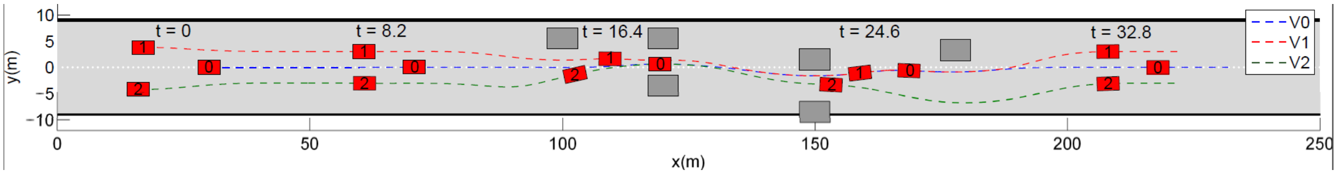


Figure 11 Convoy driving with obstacle avoidance capability

The architecture has been implemented in the two Cybuses of INRIA. We have performed the car-following experiment (as a simplification of convoy) and the intersection crossing experiment (as a simplification of autonomous intersection management). Detailed validation results are available in Deliverable D5.2.

4.2 Perception Layer

Building up reliable and accurate knowledge of the environment—often called an environmental model or perception layer—of autonomous vehicles is a crucial requirement in order to perform automated maneuvers. In general, the environmental model comprises static and dynamic entities which need to be observed and tracked over time. As both stages, decision making and control algorithms, directly depend on the robustness and the accuracy of the environmental model, this is considered a core part in automated driving.

The perception module of AutoNet2030 represents an abstraction layer which is embedded between the physical sensors and the function layer as depicted in Figure 12.

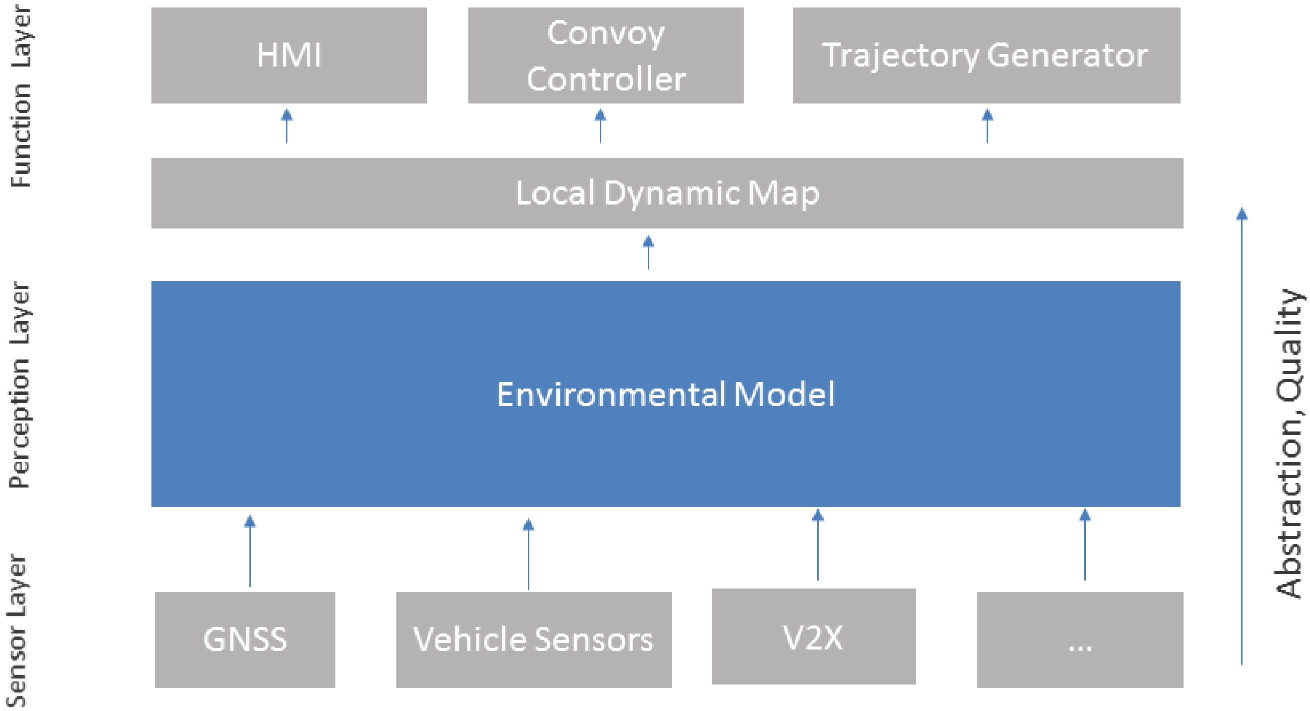


Figure 12: Perception layer as implemented in AutonNet2030

Within the AutoNet2030 project a perception module that integrates on-board perceptions sensors as well as communication sensors is required in order to safely perform use cases for high and low speed scenarios. Thus, the environmental model realizes a sensor coverage of 360 degree around the host vehicle. Furthermore, most of the sensitive areas around the vehicle (e.g. the blind spot area during lane change maneuvers) are observed by at least two independent sensors (i.e. on board corner radar as well as V2V sensor) for improved confidence in object detection or free space estimation. The perception layer has to be efficiently implemented to meet the real-time conditions during the demonstrations and optimally exploit the computational resources of the embedded computer (“central brain) used inside of the test vehicles. Finally, as AutoNe2030 deliberately uses several heterogeneous vehicle platforms, the perception layer has to be easily adoptable to different vehicle configurations. In order to address these challenges, a novel tool-based development approach has been used inside of AutoNet2030.

In the following paragraphs the three main challenges and outcomes of the implementation of the AutoNet2030 perception are introduced.

4.2.1 Multi-Vehicle and Multi-Sensor Integration with Configurable Perception Layer

In AutoNet2030 four different heterogeneous vehicle platforms have been used, that is, several SCANIA trucks, a Fiat 500, a VW Passat and the INRIA Cyber cars. Each vehicle platform has a specific sensor configuration and mounting parameters as well as proprietary sensor interfaces. In Table 1 these platforms and their corresponding sensors are presented. Moreover, each vehicle is equipped with V2V communication and a local GNSS sensor for ego positioning.

Table 1: AutoNet2030 vehicle platforms and available on-board sensors

			
• Long and mid range radar • Front looking mono camera • Two corner radars	• Front looking laser scanner	• Long and mid range radar • Front looking stereo camera • Surround view camera system	• Multiple laser scanners • Front camera

In contrast to typical state-of-the-art implementations, where a perception layer is developer for each vehicle platform individually, in AutoNet2030 only one perception has been developed. Based on the overall sensor configuration of the AutoNet2030 platforms, a unified perception subsystem has been created which is a superset of all potential sensors as listed in Table 1. By that, the whole area around the host vehicle is observed at least by two independent sensors. Thus, the complete list of supported sensor types comprises:

- A front looking radar with near and far range
- Two corner radars to observe the rear and blind spot area
- A vision based surround view system
- A front looking Mono camera system
- A front looking Stereo camera system
- A front looking Lidar system
- A V2V sensor for 360° coverage

Based on the platform the AutoNet2030 perception is used on, only a subset of the fusion capabilities is activated via configuration. Moreover, the particular sensor configuration (e.g. mounting position, statistical properties) may be different. By that approach, only one tracking/data fusion system needs to be developed. The configuration can happen during run-time and is vehicle specific.

In order to realize that challenging task in a efficient manner, the configurable multi-vehicle/multi-sensor perception has been developed in a prototypical way by using the high level programming languages C# and C++ in combination with the probabilistic data fusion SDK Baselabs Create. This SDK naturally leverages an incremental development process where each sensor can be added one by one until the final 360 degree configuration is reached.

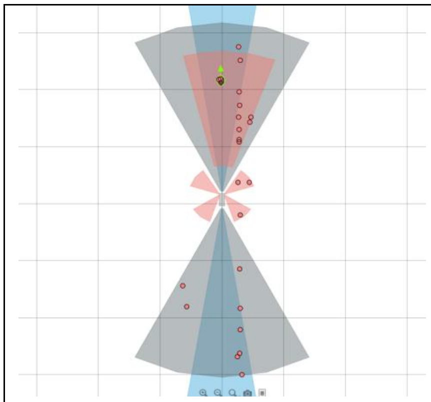
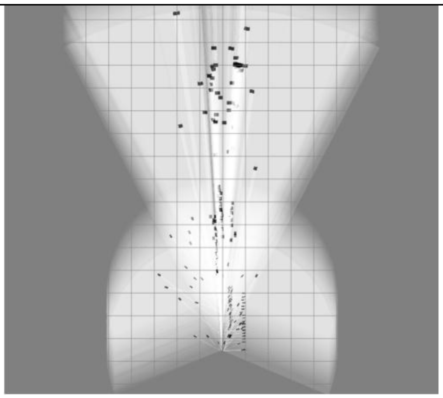
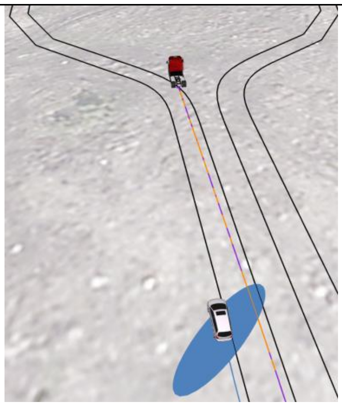
The resulting environmental model was tested and validated, both with recorded and online measurement data for each vehicle platform. Moreover, is has been used in the simulation framework Webots [**10**].

4.2.2 360 Degree Perception with Communication

Besides using local on-board sensors as an input for the perception subsystem, AutoNet2030 strives to use information received via V2V communication as an addition sensor source. By that, the following added value coud be demonstrated:

1. By using the CAM of the 802.11p stack as an input to the perception, the range of the surveillance area around the host vehicle could be improved for a horizon of up to 300m under demonstration conditions. Moreover, temporarily occlusion of objects through shadowing – as it might happen with on-board sensors – has been addressed. This is especially important for doing cooperative controlling maneuvers within a larger convoy across multiple vehicles.
2. In contrast to on-board sensors, object initialization by CAMs allows to properly set most of the initial states of a tracking hypothesis from the beginning. Besides dynamic states such as velocity and acceleration, this includes static parameters such as width and length as well.
3. In addition of using the CAMs for the modeling of dynamic road users, AutoNte2030 also investigated the potential of sharing sensor information across vehicles in a vehicle-independent way. Therefore, the Cooperative sensing services (CSS) has been developed.

Table 2: Algorithmic components of AutoNet2030 perception system

		
Multiple object tracking	Free space estimation	Ego localization

The implementation of the AutoNet2030 perception system comprises three main algorithmic blocks (compare Table 2) which serve different purposes.

The multiple object tracking (MOT) is responsible for estimating the dynamic (e.g. position, velocity, acceleration) and static parameters (object width and length) of dynamic road in a probabilistic manner. Moreover, it additionally estimates a confidence value representing the existing of an object on the road and thus directly handles sensor inaccuracies such as clutter or missing detections. In AutoNet2030, the MOT is driven by all exteroceptive on-board sensors and the CAMs received via V2V. The output of the MOT is a so-called track list which represents all confirmed vehicle hypotheses around the host vehicle. As the AutoNet2030 perception strives to abstract particular sensors and vehicle parameters, the track list is provided within a constant update rate and has a similar structure and content on all AutoNet2030 vehicle platforms. By that, the upper layers such as the convoy controller and the HMI could be implemented in a sensor agnostic way.

In contrast to the track list provided by the MOT, which is mainly used for performing longitudinal control between vehicles, the purpose of the free space estimation is to represent static environment. Therefore, the AutoNet2030 perception is equipped with a occupancy grid algorithm that is driven by lidar and radar raw measurements to estimate the drivable and non-drivable area around the vehicle. The free/occupied information is stored within a geo-referenced grid and then transferred via a dedicated V2V message to other AutoNet2030 users. The receiving AutoNet2030 entities, which estimation an occupancy grid own with their own sensor, then combine both grids to get an improved confidence value for free space. Finally, this information is used within the upper layers of the AutoNet2030 architecture to make decisions about the initialization or cancelation of maneuvers such as lane changes.

Both, the incorporation of CAMs within the MOT as well as the CSS, required the on-board perception to have a reliable knowledge of the host vehicle's ego position with respect to an absolute coordinate system such as WGS84. Therefore, a third data fusion component, the ego localization system, was implemented within the AutoNet2030 perception. Basically, the localization system is an implementation of a loosely-coupled GNSS/INS fusion scheme of absolute GNSS positioning data and in-vehicle odometry data such as velocity, yaw rate and acceleration. Moreover, a vehicular motion model was applied in order to stabilize the sensor measurements and account for errors such as biases (yaw rate) and scale factors (velocity). Furthermore, the estimated position, velocity and heading information are passed to the local 802.11p stack to provide the input for the CAM generation.

4.2.3 Adoption to an Embedded Platform

The configurable perception system described in section 4.2.1 was initially developed and tested based on the prototyping middleware vADASdeveloper which is a plugin for Microsoft Visual Studio. Consequently, it was a standalone GUI application running under the Windows operation system with support of the .NET runtime environment. At the beginning of the project, this rapid-prototyping approach was beneficial due to the powerful debugging support and rich domain-specific visualization capabilities of the aforementioned tools. However, it turned out that for the highly-integrated hardware system at which the AutoNet2030 project targets at, this approach was not optimally suited. In AutoNet2030, a single computer platform that runs the Linux OS with enhanced real-time-capabilities was foreseen. In this “central brain” unit, every subsystem was supposed to run in an isolated and resource-controlled environment via support of application containers.

Hence, the configurable perception was transformed from a Window GUI application to a headless embedded version which was compatible with the Mono .Net implementation available under Linux. This transformation mainly involved a change of the underlying middleware from vADASdeveloper towards a lightweight LCM-based infrastructure [**3**]. The perception module was encapsulated as a self-contained Docker image [**2**] and deployed to the vehicles of the partners. The configuration (e.g. platform selection, parameter optimiziation) was still possible by changing command line parameters.

4.3 V2X Communications

The body of work carried-out in the context of the project involves both the implementation of innovative communication services and for the actual demonstrated system but also the performance evaluation of relevant algorithms, carried-out using simulation tools. Both are briefly presented here.

4.3.1 Communication requirements for cooperative automated driving

The first generation of V2X communication (1G-V2X) addresses mainly road safety and traffic efficiency for manually-driven vehicles. Typical applications include obstacle warning, road works information, in-vehicle signage, traffic light phase assistance, and others. The use cases for autonomous driving demand new requirements. New *functional requirements* include [**18**]:

Additional vehicle status data. In 1G-V2X, every vehicle broadcasts periodic safety messages to inform neighbours of its position, speed, heading, and other parameters. Autonomous vehicles need to include additional data fields carried in the periodic messages for the convoy driving and cooperative lane change use cases, such as their maneuver intentions.

Convoy management. In 1G-V2X, a vehicle communicates with vehicles and roadside stations in its neighbourhood, or located in a specific geographical region, also called a relevance area for safety information. Opposed to this `open group' concept without an explicit membership, a convoy represents a `closed group' where a vehicle needs to become group member to participate. In order to create and maintain convoys, as well as to coordinate decentralized maneuvering negotiations, new fault-tolerant mechanisms for group management are needed.

Maneuver negotiation. In autonomous driving, vehicles may actively need to reserve road space for lane change maneuvers. Differently from the distribution of periodic or event-driven safety messages for 1G-V2X, a reservation requires a negotiation among the involved vehicles to request and acknowledge the maneuver. This exchange enforces optimal and safe trajectories for the cooperative vehicles and minimizes their collision risk.

Intersection management. 1G-V2X is limited to the periodic broadcast of static and dynamic information of intersections, i.e., to distribute the intersection topology and traffic light information, enabling use cases such as green light optimal speed advisory. It also allows for requesting and changing the status of traffic light control systems for priority control and preemption of road traffic. With autonomous driving, communication for intersection management is extended to allow for more detailed information of the intersection geometry and to assign priorities to incoming vehicles, potentially displacing traffic lights.

Cooperative sensing. Communication allows for the exchange of locally acquired sensor data from the radar, camera, and other sensors. The captured data from the local sensors is aggregated into a list of detected objects along the road, such as obstacles, vehicles, pedestrians, that can be exchanged with neighbouring vehicles. The cooperative sensing increases the sensors' field of view to the V2X communication range and enables cooperative perception among vehicles. In 1G-V2X, the aggregation level of sensor data is much higher and messages only carry a coarse event classifier and relevance area. Instead, the cooperative sensing use case requires the exchange of highly detailed information about the detected objects.

In addition to the functional requirements, specific qualitative *performance requirements* for cooperative autonomous driving include:

High message rate. In 1G-V2X, vehicles periodically broadcast safety messages with an interval between 100 ms and 1 s, where the rate within these limits is controlled by the dynamics of the generating vehicle and the load on the wireless channel. In contrast, the small inter-vehicle distance among autonomous vehicles requires the use of a high and fixed broadcast frequency with a timeliness guarantee on the information that autonomous vehicles possess about their neighbours. These requirements target autonomous vehicles to have a complete and up-to-date environmental model, which allows them to coordinate maneuvers in a safe manner.

Data load control. The small inter-vehicle distance and the corresponding high vehicle density lead to a higher data load in the network. This is even amplified by the high message rate and by additional data load for the exchange of control messages. In order to control the amount of data traffic in the network, an efficient utilization of the available frequency spectrum, an effective prioritization of messages by the decentralized congestion control (DCC) function, and a strict control of the forwarding operations are required.

Low end-to-end latency. The end-to-end latency is mainly composed of the delay to gather data from local sensors, the processing delay in the protocol stack, and the transmission delay over the wireless link. The end-to-end delay also includes the delay induced by the security mechanisms (generation and verification of signature and certificate, respectively) and by the queuing delays in the DCC function. In 1G-V2X, the latency requirements for critical road safety applications are set to 300 ms [ETSI TS 102 539-1]. In autonomous driving use cases such as convoy driving, the latency requirement is more stringent due to the smaller inter-vehicle distance between vehicles and also to ensure the string stability of large convoys.

Highly reliable packet delivery. The requirement for a reliable exchange of information is more critical than in 1G-V2X, since a lost or erroneous message might cause a malfunction of the vehicle control algorithms and create a safety risk.

Both functional and performance requirements impose demanding challenges on the V2X communication system. In the AutoNet2030 we have proposed enhancements of 1G-V2X to meet some of these challenges.

4.3.2 Message Extension for Cooperative Automated Driving

Cooperative Awareness Message Extensions

The Cooperative Awareness Message (CAM) is a 1G-V2X standard message in Europe [ETSI EN 302 637-2] for the exchange of ego-vehicle data among nearby cooperative vehicles using vehicle-to-vehicle communication.

Autonomous driving requires the exchange of additional vehicle status data not supported by the standard CAM to date.

AutoNet2030 extended the CAM with additional fields to support the convoy-driving and cooperative lane change use-cases. In particular, additional data fields were added to the CAM to carry vehicle control data such as the vehicle's target speed, target acceleration and target trajectory while preserving standard compatibility as far as possible.

The transmission frequency of the extended CAM in AutoNet2030 was fixed to 10Hz. for vehicles driving in a convoy or performing a cooperative lane change. A fixed transmission frequency eases the fusion of CAM messages with local perception sensors for receiving vehicles, despite an inherent loss probability of packets on the wireless medium.

A higher transmission frequency typically bring along a higher data load. AutoNet2030 deals with higher channel loads by distributing the CAM transmissions over multiple channels and limitation of message size by using optional data fields to the absolute minimum.

Convoy Control Communication Service

The *Convoy Control Communication Service* (CCCS) supports the exchange of information messages among cooperative vehicles in the convoy driving use case and satisfies the functional requirement for convoy management (see Section 4.1).

The transmission frequency of convoy messages is dynamically adjusted depending on the convoy properties and the traffic conditions.

The messages exchanged among convoy vehicles via the CCCS enable each vehicle to maintain a local graph whose nodes are the convoy members and the edges represent the dependence of the vehicle dynamics. A decentralized vehicle control algorithm performs the cooperative maneuvering, adjusting the vehicle lateral and longitudinal dynamics to keep a balanced formation and performing lane changes as required [**7**].

The message types offered by the CCCS to convoy members are:

Join/leave convoy. A *join request* is a single-hop broadcast message sent by an approaching vehicle, which detects a convoy and requests to become a convoy member. Similarly, a convoy vehicle which decides to abandon it (e.g., when it reaches its destination) will broadcast a *leave request* to inform its neighbours of its intention.

Lane change. A *lane change* message allows convoy vehicles to change their lane within the convoy. The message is broadcast by a convoy vehicle to inform its neighbors of a planned lane change. This way, the convoy members in the destination lane will adjust their positions to give space for the incoming vehicle.

Modify local graph. As a result of a lane change or a new vehicle entering the convoy, a vehicle may update its local graph. In this case, the new graph is broadcast to its neighbours by means of a '*modify local graph'* message. The neighbour vehicles then modify their own local graphs accordingly; thereby ensuring the consistency of the graphs among all the convoy members.

Cooperative Lane Change Service

The cooperative lane change service (CLCS) is supporting the negotiation of lane changes up to 10 seconds in advance. The CLCS is a service in support of the convoy controller and divides the negotiation in three phases: the search, preparation and execution phase:

Search phase. In this optional phase, the CLCS of a subject vehicle – a vehicle initiating the lane change – supports the convoy controller in finding an appropriate target vehicle for a lane-change or highway merge negotiation. Due to sensor occlusion and limited 1-hop wireless transmission range, a subject vehicle may not be aware of so-called target vehicles. The CLCS supports the subject-vehicle in finding the appropriate target vehicle using multi-hop broadcast communication.

Preparation phase. In the preparation phase, a subject vehicle and target vehicle are periodically updating their lane change intention until either vehicle aborts or the target vehicle confirms the requested lane-change gap to the subject vehicle. Either message is periodically retransmitted until acknowledged by the peer vehicle.

Execution phase. The lane change maneuver is performed during the execution phase. Once this phase is finished, the period broadcast of the lane change intention is stopped by subject and target vehicle.

Cooperative Sensing Service

The cooperative sensing service (CSS) is a service for exchange of free space estimated by the perception module. The service is supporting the cooperative lane change use-case and aims cover the blind spots of vehicles, in particular trucks with large trailers, by the cooperative exchange of occupancy grid data.

The occupancy grid data, depending on the cell size, grid size and bits per cell, may quickly grow beyond the maximum transfer unit of ITS-G5. In AutoNet2030, we applied a real-time encoding and decoding algorithm to fit the occupancy grid data inside one packet while limiting the overall message processing time.

Nevertheless, the size of a cooperative sensing message (CSM) is order of magnitude larger than standard CAM. AutoNet2030 decided to broadcast CSMs on a service channel to offload the control channel and avoid competing with high-priority CAMs for channel access.

4.3.3 Smart Forwarding Algorithms for Cooperative Automated Driving

In order to forward packets among cooperative vehicles in a multi-hop fashion by means of V2X communications, ETSI has defined the GeoNetworking protocol. GeoNetworking is based on geographical routing and makes use of the geographical positions of the cooperative vehicles to find forwarding paths from source to destination.

Several forwarding algorithms (both unicast and broadcast) have been defined in the GeoNetworking standard ETSI EN 302 636-4-1 [**19**]. However, these algorithms are designed for safety-oriented vehicular communications, generally considering the single-hop broadcasting of information and the multi-hop distribution of information within a geographical area. They do not consider the specific requirements of cooperative autonomous driving systems (Section 3.1). For instance, cooperative autonomous vehicles driving in a platoon or convoy form a closed group of vehicles which have a very short distance among them, thereby requiring extremely low-delay and high-reliability communications to ensure their safe coordinated manoeuvring.

In the AutoNet2030 project, we have proposed two new forwarding algorithms which aim to enhance the performance of standardized GeoNetworking routing protocols in order to satisfy the requirements of cooperative autonomous driving. In particular, we introduced *Greedy Broadcast Forwarding* and *Greedy Multicast Forwarding* [**20**], designed to allow fast and reliable information dissemination within vehicle groups, such as platoons and convoys.

Greedy Broadcast Forwarding (GBF) is a novel forwarding algorithm designed to enable reliable and low-latency communications in AutoNet2030. A typical scenario where this algorithm may be used is in information dissemination within vehicle groups, where a group member distributes relevant information among all other group vehicles. Similarly to the GeoNetworking protocol, GBF assumes that all group vehicles know their own position (e.g., by means of GPS) and have a location table with the ID's and positions of the neighbouring vehicles. The main features of the GBF algorithm can be summarized as follows:

1. The source node selects a next hop by Greedy Forwarding (GF), i.e., the neighbour with the highest forward progress towards the destination, and forwards the packet to this node. The destination is defined as the geographical position at the opposite edge of the geographical area. The selected GF next hop will in turn forward the packet by GF as well. If no next hop is found, the packet is discarded.
2. Other nodes within the communication range which overhear the transmission also process the packet. At the same time, they start a retransmission timer whose value is proportional to their distance to the GF next hop (the exact value will depend on the scenario). This way, in the case that the GF next hop does not receive the packet, other nodes are able to retransmit it.
3. If a node overhears a packet for the second time, it means that the packet has already been retransmitted by another node. Then, the node cancels its retransmission timer and discards the packet.
4. If the retransmission timer expires and no further transmission has been observed, it means that the node is responsible to retransmit the packet. The node calculates its GF next hop and forwards the packet by GF.

Using this algorithm, the packets are broadcasted to all nodes within the geographical area with a low delay, since packets are forwarded to the node within communication range with the highest forward progress. Furthermore, vehicles overhearing a packet transmission exploit the information about their position compared to the GF next hop in order to set their retransmission timers to much lower values compared to generic broadcast routing protocols, such as contention-based forwarding, which sets the retransmission timer as a function of the distance to the previous forwarder only. This mechanism allows for very fast retransmissions in case of a collision in the GF transmission, which result in a low end-to-end delay even in unfavourable scenarios.

Greedy Multicast Forwarding is a similar forwarding designed for a multicast scenario, i.e., when the messages are directed only to a subset of the vehicles present in a given area (e.g. a convoy). The implementation details of both forwarding algorithms and their performance evaluation with respect to the standardized GeoNetworking protocol can be found in [**20**].

4.4 Human-machine Interface

In light of the increased awareness capabilities provided by the aforementioned cooperative V2X messaging, HMI systems are now required to leverage information beyond the scope of the ego-vehicle perception system and provide it to the driver with the maximum possible clarity. This Section describes the way that the AutoNet2030 has address this challenge introducing an innovative dual-display approach to HMI visualizations.

4.4.1 Design Requirements for Ergonomic HMI

The approach for AutoNet2030 User Interaction [**21**] has been to start from a reference scenario consisting of partially autonomous vehicles, i.e., to start from the lower SAE automation levels. A multidisciplinary work has been carried out involving engineers, software and hardware experts and interaction and cognitive HMI experts. The objective was to achieve a user interface considering both driver comfort and easiness of use. The peculiarity of AutoNet2030 was that we had to consider different automation levels, including the manually driven vehicle (level 0). CRF as main co-responsible of the HMI task, had the convoy driving as target scenario, therefore, the HMI work focused on the information flow which has to be provided when driving within a convoy.

1. The related functional Use Cases (joining, merging, lane change, leave) were analysed from a user/driver point of view. Each Use Case has been broken down into a flow of events and actions the user needs to cope with, while driving. This analysis has been conducted considering both automated and manual driving conditions (all Use Cases correspond to driving conditions, none of them to stationary conditions). According to the role of the user, there are Use Cases invisible to the user: system actions and communications with other vehicles that do not affect the driver situation awareness, are not useful for the driver or could be distracting
2. Informative Use Cases: system information or advised manoeuvres the driver needs to be informed about
3. Interactive Use Cases: the user will be able to interact with the HMI system

4.4.2 Implementation

The HMI system has been implemented into two modules in line with AutoNet2030 architecture. The HMI modules consist in a HMI support interfaced to the other project components (e.g. perception, controller, etc.), and the HMI management which handles connectivity and messaging towards and from the HMI device peripherals (applications on the devices). The two modules are described below.

4.4.3 HMI Support module

This module (Figure 13) collects and organizes the bidirectional traffic between interested applications of the internal system and the HMI module. Moreover, it decouples the HMI application logic from any OEM-specific characteristics.

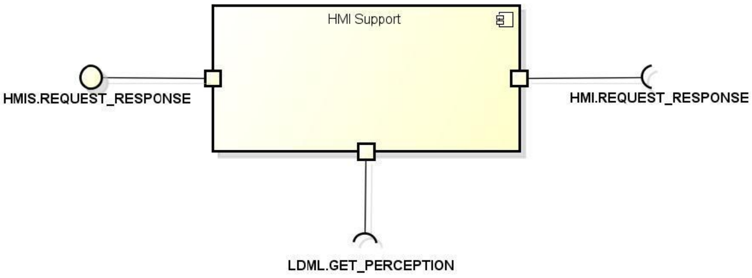


Figure 13 – Design and associated interfaces for the HMI Support module

Essentially the information sent towards the HMI to drive the projections over those devices is a combination of data coming from the Controller and the Perception module.

Therefore, in terms of the involved interfaces:

- the Maneuver Controller interface receives input data related to manoeuver suggestions;
- the Perception interface receives input data related to road perspective and surrounding vehicles (and the (core module of the) HMI System.
- the HMI Management interface provide output data

The Maneuver Controller interface (HMIS.REQUEST_RESPONSE) relies on Lightweight Communications and Marshalling (LCM) [**3**] messages to accommodate velocity profile (current vs. target speed), lane-index/lane-change information and convoy status, as estimated by the Maneuver Controller.

Information coming from the sensors i.e., Perception module, such as the number and coordinates of tracked surrounding vehicles and the information of the current lane occupied by the vehicle, are retrieved by the HMI Support module (through the LDML.GET_PERCEPTION interface from the digital map LDM sub-component); thus, the system is enabled to provide more informed HMI visualizations.

The last interface (HMI.REQUEST_RESPONSE) accommodates an internal communication channel with the HMI Management module the functionality of which is described below.

4.4.4 HMI Management Module

The second module represents the core of the HMI functionality. It analyzes the content of incoming requests and provides the driver with the necessary notifications, interaction with the system and/or maneuvering advising. The adopted design approach allows to support more than one display using a single HMI instance (in the vehicle platform) regardless the current driving mode. It includes three sub-modules:

- the HMI Management service handles the request-response functionalities, composing the messages for the devices
- the HMI Scheduler does a priority-based scheduling of output messages
- the Device handler manages the WiFi connections to the devices

The modules, reported in Figure 14, are described hereafter in detail.

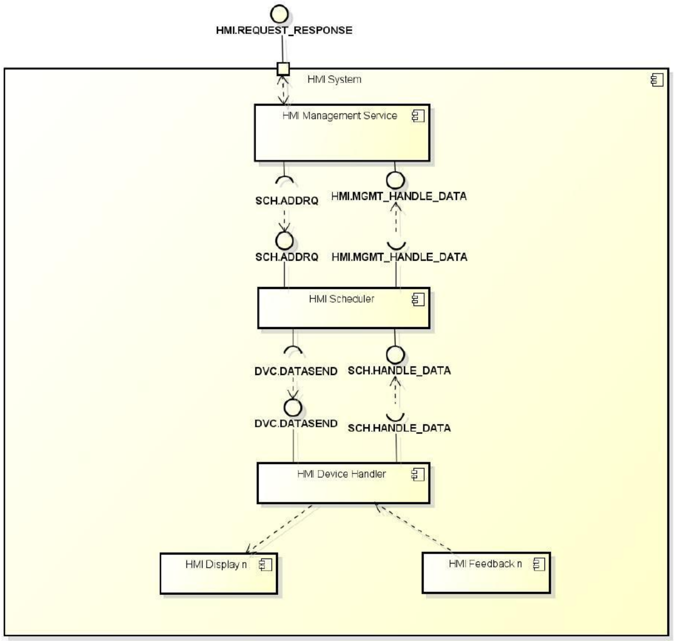


Figure 14 - Design and associated interfaces for the HMI Management module

The HMI Management service provides the HMI request-response functionalities. It analyzes the input received from AutoNet2030 components, and based on this input it fills in predefined JSON templates with the needed content to be displayed on the screen. The management of the resulting JSON messages is then handed over to the scheduler through the HMIS.REQUEST_RESPONSE interface. In brief, each single JSON message specifies the content displayed in each sub-area of the screen, as outlined in the following section, so it represents 1:1 a specific screen-shot that should be displayed in a specific time instant. The optimization of updates on the screen (i.e. to transmit only changes to the screen, avoiding repetitions of the same content) is handled by the scheduler.

The HMI Scheduler has been designed to carry-out priority-based scheduling of the (incoming) HMI messages. It can therefore support a logic i.e., a way to prioritize the JSON messages with respect to a considered set of their attributes. It does two kinds of updates: an asynchronous, event based update to display new content on the device (transmit only the changes) and a periodic refresh of the content where the entire screenshot (i.e. the whole JSON information content) is updated.

The Device handler manages the bi-directional WiFi connection with the display-devices in a transparent way for the upper level. The usage of the wireless channel to access the Android device relies on well-established procedures (i.e., through a network SSID applying the appropriate permissions). Then, the component serves two main purposes regardless the number of the final display devices:

- To send to the displays any HMI request and relevant content to be projected according to the designed AutoNet2030 wire-frame (detailed in the next Section).
- To receive feedback/inputs from connected User-Interface devices (e.g., Android).

4.4.5 Dataflow from an HMI Standpoint

Figure 15 gives an overview of the data flow from the HMI system perspective. LCM format [**3**] is used to define the majority of the exchanged messages (as is the case for the MC directives received by the HMI). Since the Maneuver Controller component is primarily intended for vehicle-automation capabilities (requiring high MC output rate), a Filtering module is present between MC and HMI components. Its aim is to reduce and harmonize the number of directives generated by the MC component, and provide an output rate compliant to the human ability to perceive a message. To constantly check the Driver Conformance to directives provided by MC, a dedicated module is running in parallel to the HMI. This module compares the directives from MC with the vehicle dynamic data, to determine whether the driver is following correctly the system’s directives.

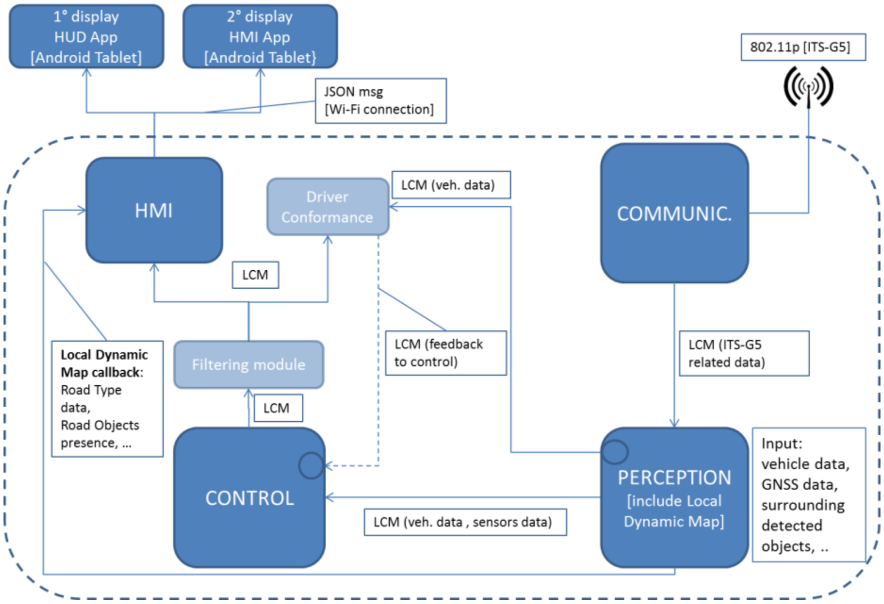


Figure 15 - Modules interaction in the AutoNet2030 architecture

4.4.6 Wire-frame of Displays

To obtain the needed wireframe of displayed content, a break-down of the displays into a number of sub-areas was made [**21**]. The objects or text messages are projected in specific sub-areas (specified by the aforementioned JSON template), to advise/inform the driver in line with any road/vehicle event. Figure 16 shows the AutoNet2030 wire-frames for the HUD and secondary device. For each display used (HUD, secondary device), a layout has been studied and created to cope with the different displays content, in order to have a consistent organization, helpful in speeding up the driver understanding.

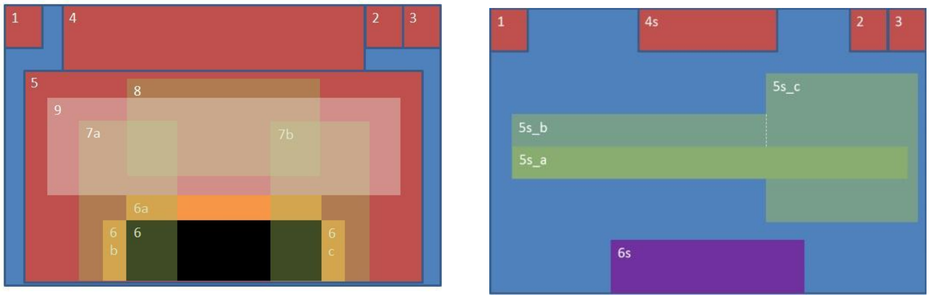


Figure 16 Sub-areas of HUD wire-frame (left) and secondary device (right)

Certain pieces of information are projected over the defined wire-frame of the devices in a consistent way so that the driver can quickly understand. The effort was to keep some basic information projected in the same way across the screens.

HUD

The head-up display is designed to project directives to the driver as well as the main display elements (i.e., road, arrows, vehicle shields etc.). The various pieces of information that is projected over the sub-areas of Figure 16(left) as well as some indicative encoding (e.g., selected colors) are presented in the following table.

Table 3: Wire-frame specifications for the HUD

Sub-area ID	Projected Information
1	Driving mode (M: manual, A:automated)
2	Convoy/Platoon motion (C/P letter)
3	Local Cooperative Area icon
4	Target speed assistant
5	Road with one/two/three lanes
6	Ego vehicle in the color of the current modality (*e.g.,* automatic in blue)
6a	Frontal shield colored in line with the situation criticality
6b/6c	Left/Right lateral shield
7a/7b	Left/Right arrow for left lane change
8	Preceding vehicles
9	Auxiliary sub-area

The central sub-area (no. 5) presents the road and optionally visual objects. The selection of the road type for the background reflects the digitalized map available through LDM and the map-matching procedure over this digital map. Vehicle-status information (e.g., manual or automated driving mode) and directives/informative messages appear in the top horizontal area. The projection in sub-area 4 is dedicated for the visualization of current vs. target speed information, with a graphical solution of a "target speed assistant" widget. The motivation of this approach is twofold: (1) text messages have to be omitted from the HUD for clarity reasons; (2) Typically, the controller directive to the driver consists of a suggested speed and lane to follow so a dedicated area to present this advice is needed. The arrows (of sub-areas 7a, b) will be projected (and hidden) according to the Maneuver Controller decisions in a way that allows the proper execution of lane change maneuvers.

Secondary Device

A similar wire-frame configuration has been introduced for the Android device which -in this case- maintains an I/O capability *e.g.,* touch screen. The sub-areas of Figure 16 (right) that project different content compared to the HUD (Figure 16 left) are summarized in the following table:

Table 4: Wire-frame specifications for Secondary device

Sub-area ID	Projected Information
4s	AutoNet2030 Logo
5s_a	Short text explanation (no directive message) of requested maneuver
5s_b	Short text explanation (no directive message) of requested maneuver
5s_c	Graphical elements which comes with written feedback (e.g., traffic jam icon)
6s	Interactive area with a virtual central button

The Android HMI application, installed in the secondary display, can cope with different driving modes and operate both in-parallel with the HUD or as a standalone visualizer. The use of the interaction button of sub-area 6s enables the user to switch the driving mode, given that safety conditions are met.

Alternative Configuration

This wireframe is used in vehicles operating under autonomous mode. In this case the expectation is that only an Android device will be present and acting as the Secondary Display (no HUD activated). This Android device can be either an external tablet mounted on the dashboard or an Android-compatible screen of the vehicle's (entertainment) system. The aim of this alternative configuration is mainly to inform the passengers/driver of AutoNet2030 automated prototypes, about the directives that the system (i.e. Maneuver controller component) has taken and executed during the cooperative maneuvers.

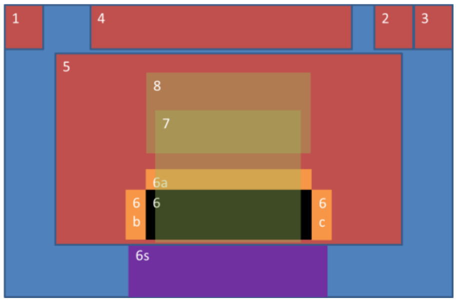
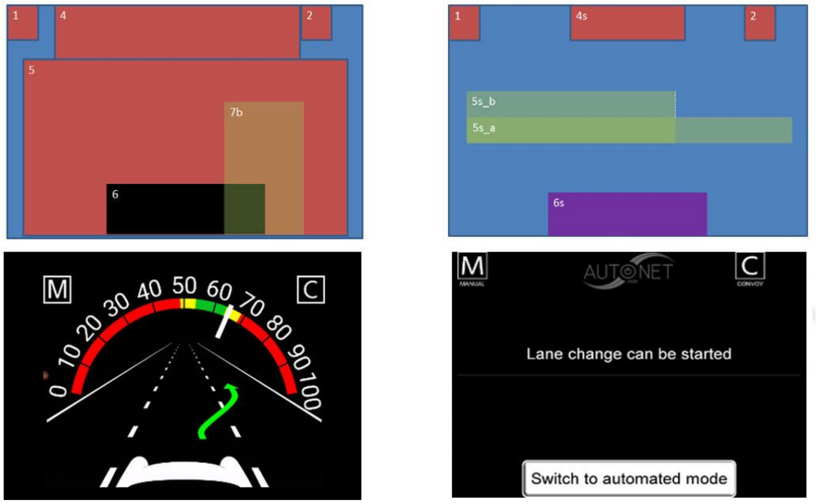


Figure 17 – Sub-areas of the Android device screen under the alternative configuration

The proposed wire-frame for this case is essentially a combination of the previous two (see Figure 17). The central area of the screen (i.e., all sub-areas included in sub-area 5) reproduces the HUD projections but only for informative reasons (rather than providing directives). It can include the target speed assistant (as for the HUD standard configuration) while the sub-area 4 shows short text messages to complement the passenger's awareness about road/vehicle events.

4.4.7 Visual Objects and Text-messages Vocabulary

Following the work of the wire-frame design, we have created a complete set of display objects that help us clearly communicate to the driver any appropriate advices/information as suggested by the Maneuver Controller. In particular for the Target speed assistant of sub-area 4 during the development phase we have modified the first approach defined during the design phase, to obtain a larger visualization of the target vs. current speed information for the driver, without the usage of textual advice. In Figure 18 we consider a lane-change scenario and present the corresponding sub-areas that will be 'involved' out of the proposed wire-frame. We also show the two corresponding snapshots (of the HUD and Android visualizer applications) illustrating how the AutoNet2030 display objects are used.



Lane change can be started

Figure 18 - Sub-areas affected and visual objects displayed in response of a specific directive (HUD: left and Secondary Display: right).

The adopted general strategy is to inform the driver projecting text messages in the secondary display that explain the rationale (i.e., the reason) behind each directive (visualized in the HUD). For the sake of clarity, specification work has been carried out to define a set of rules that clearly prescribe the syntax of the projected text messages and the lexicon that is accordingly in use.

5 Contribution to ITS standards

As opposed to the vast majority of ITS research project, AutoNet2030 had an active presence in standardization organisations (*e.g.*, ETSI) and made significant contribution to upcoming vehicle-automation standards. These achievements are briefly presented in this section.

Related deliverable:

D6.1 Report on the performed cooperative vehicle automation related standardisation activities and standardisation status (*This deliverable describes the standardisation activities performed during the project as well as the way that the AutoNet2030 contributions to the ETSI ITS standardization work have advanced the related ITS standards*)

Since AutoNet2030 has been developing connected and cooperating automated driving technology, the interoperability of interactions among automated vehicles is essential. Achieving such interoperability necessitates turning the specified communication interfaces into actual standards, which is achieved through interaction with the relevant standards development organisation. The most relevant standards development organisation for the scope of AutoNet2030 work is the ETSI ITS committee, which has the aim of standardising cooperative ITS communications. The AutoNet2030 consortium members are very active in the work of this committee, with the following partners being ETSI members and active contributors to the work of ETSI ITS: Hitachi Europe, TU-Dresden, Armines, and BroadBit. In fact, the Applications working group (WG1) is chaired by Lan Lin from Hitachi Europe, and the Networking working group (WG3) is chaired by Dr.Andreas Festag from TU Dresden/Fraunhofer IVI, both of whom have taken part also in the AutoNet2030 work.

The AutoNet2030 contributions to the ETSI ITS standardization work have advanced the related ITS standards from the “Day-1” set of driver warning applications towards the next level; i.e. towards the transactions for automated driving facilitation. The needed standardisation work items for automated driving support have started to be initiated already in the first project year, with the starting of ‘Platooning’ and ‘Cooperative ACC’ work items. This work has continued in the second and third years with the contribution of relevant AutoNet2030 specifications. The following table shows the assignment between AutoNet2030 specification topics and the relevant ETSI ITS work items, where the specifications shall be contributed.

Table 5: List of standardisation work items in ETSI TC ITS where AutoNet2030 is contributing

AutoNet2030 specifications	ETSI TC ITS Work Items
AutoNet2030 use cases	TR 102 638: use case definition
CAM extension for cooperative automated driving applications	EN 302 637 – 3: CAM basic service
Cooperative lane change service	Potential new WI proposal
Convoy control service	TR 103 298: Platooning pre-standardization TR 103 299: Cooperative ACC pre-standardization
Cooperative Intersection Control Service	TS 101 539-2: Intersection collision risk warning
Cooperative sensing service	TS 103 324: Cooperative Observation Service
Cooperative RTK Positioning	TS 102 890 – 3: Position and Time
All message format specifications	TS 102 894- 2: Common Data Dictionary
Reliable Basic Transport Protocol	ETSI EN 302 636-5-1: BTP
Extended GeoNetworking	ETSI EN 302 636-4-1: GeoNetworking

The AutoNet2030 partners involved in contributing to the above-listed work items are committed to continue this work also after the conclusion of the project, till these contributions reach the maturity of becoming published ETSI standards.

6 Dissemination and Exploitation of Project Results

The project has accomplished an impressive set of publications, talks, participation in exhibitions and standardization bodies as well as presence in well-known ITS events. Together with the successful organization of a highly-visible final event, the overall set of activities indicate the significant outreach of the AutoNet2030 work and lays the ground for successful exploitation and maximization of the project's impact. A brief mention of these activities appears in the current Section.

Related deliverable:

D6.1 Report on AutoNet2030 dissemination activities (*This deliverable details the various dissemination activities along the project's lifetime*)

6.1 Summary of Dissemination Activities

The consortium considered the broad dissemination of the project’s results as of paramount importance. A concise dissemination strategy was established from an early phase in order to maximise the visibility (even outside European boarders!) and successfully diffuse the AutoNet2030 achievements. The corresponding strategy prescribed how to effectively reach the specified target groups through the AutoNet2030 dissemination channels. Those channels were depended on the stage of the work progress. While at the early stages of the project the dissemination was concentrated on presentations of the idea and AutoNet2030 concept, at later stages the dissemination task focused on presenting the achieved developments and results.

Table 6: An indicative subset of AutoNet2030 dissemination activities

Dissemination channel	Cumulative number across the project’s lifetime
Papers in conference proceedings	18
Journal/magazine articles	2 (in the prestigious IEEE Communications Magazine)
Bok chapters	1 (in a Springer book on Automated Driving)
Presentations	>40 (in conferences, congresses and ITS events)
Interviews	1 interview on vehicle convoys at the prestigious IEEE Spectrum magazine
Other activities	• Successful Demo and Workshop (Final Event) of high visibility (more than 70 attendees) • AutoNet2030 entry in the 2015 and 2016 EUCAR project book
Best paper awards	2 (ITS world Congress and IEEE Intelligent Vehicles 2016)
Best session award	1 (ITS World Congress)

Details about the dissemination strategy and relevant internal procedures are described in D6.2 (while in D6.1 the AutoNet2030 standardization activities are reported). Partners disseminated project ideas and AutoNet2030 advances at relevant conferences and other events in the area of Intelligent Transportation Systems achieving an impressive number of 62 dissemination activities (see Table 6 for a for indicative numbers). It is those activities that ensured the outreach of the AutoNet2030 achievements to the ITS community and therefore, contributed decisively to the project’s great success.

6.2 Directions for Project Exploitation

To maximize the impact of the AutoNet2030 achievements and effectively pave the way for follow-up activities a systematic plan of next-steps has been devised. This plan covers all three general directions corresponding to the different interests and profile of the AutoNet2030 partners. Those directions are briefly presented in the following points

- *Industrial exploitation*: The focus of industrial partners lies on the process to improve certain parts of the AutoNet2030 system and after exhaustive testing, include them in the internal research-and-development cycles. In this way, they can enhance their available toolkit and improve the performance of pre-production phases. Attempts to further commercialize selected parts of the AutoNet2030 work depends on its readiness level and can be in certain cases considered taking into account analysis of the involved costs. Relevant follow-ups are planned for V2X modules among the AutoNet2030 OEMs and industrial partners.

- *SME exploitation*: The partners of this business category typically seek to (collaborate further with well-known and pioneer industrial stakeholders and) receive more detailed feedback and testing results on their modules included in the project. Furthermore, extensions for the relevant software modules are planned in order to include specific use-cases that have been observed during the AutoNet2030 experimentation. Relevant follow-ups that will build upon the new and successful partnerships that have been established in the project are planned for the perception (Baselabs) and positioning (Broadbit Energy Technologies) systems.

- *Knowledge exploitation*: universities and research institutes will seek to draw on the AutoNet2030 results to enrich their expertise, strengthen their network of collaborators and identify new research directions. Research proposals and future collaborations are expected to rely on a) the interaction with ITS stakeholders and vehicle-automation experts that the project required b) the technical achievements that the project produced. The expertise gained can be used for university courses or be combined with other competences to help institutes explore promising paths of future research (such as electric vehicles or the use of big-data in vehicle-automation).

The relevant remark here is that detailed exploitation plans (mainly) for industrial partners and SMEs include sensitive information and thus, given the applied confidentiality constraints, cannot be disclosed. A subset of them is only reported in deliverables of restricted dissemination level (*i.e.*, deliverable D1.4).

7 Outlook and Conclusions

Automated driving is expected to significantly contribute to future safe and efficient mobility. Contrary to typical automated approaches that mainly focus on the host-vehicle and the related information provided by its sensory equipment, AutoNet2030 has invested more effort on a cooperative approach. Central to the latter is the need for *convergence* between sensor-based automation, cooperative V2X communications and distributed control.

The project has systematically investigated and demonstrated how cooperative communications can provide a solid basis to build decentralized control systems and at the same time enhance the capabilities of a 360^o perception layer. Moreover, the project has showcased relevant results realizing a number of demanding use-cases of both societal and customer value. Notably, those use-cases have included a *mixed fleet* of fully-automated and manually-driven vehicles "emulating" road-traffic instances to appear on the roads in the near future.

Having achieved the realization of use-cases of inherently-safe cooperative automation gives the project an excellent opportunity to shape the path towards cost-optimised and widely deployable vehicle-automation technology. Along this line and further supported by a concrete exploitation plan foreseeing both academic and industrial follow-ups, the AutoNet2030 achievements are expected to assist increasing the user-acceptance level of (future) automated driving technology.

In what follows, we conclude the Section highlighting our approach to a number of more practical issues encountered in the development of vehicle-automation technology.

7.1 Lessons learned

The consortium deems highly important to contribute to the numerous Research-and-Development cycles that seek to improve ADAS systems and lead to the era of safe, fully- automated connected vehicles. One of the means to reach this goal is the technology transfer from research to market products and consumers. Towards that direction, together with related industries, ITS stakeholders as well as the European Commission, the AutoNet2030 consortium has committed to make the relevant experiences publicly available and contribute to the evolution of vehicular automation technologies. The lessons learned during the project are presented along the main threads of the AutoNet2030 research.

7.1.1 Distributed/semi-decentralized Control of Automated Vehicles

7.1.1.1 Distributed Convoy Controller

- A multi-level iterative prototyping and validation strategy to close the gap between the theoretical design of suitable control laws and real vehicle experiments was appropriate and successful in achieving the control goals of the project.

The iterative development framework is as follows: theoretical design of the control law; validation with simulation and experiments on non-holonomic, embodied vehicles; incorporation of details of actual vehicle interfaces (e.g., LCM middleware, LDM) and physical devices in the simulation framework through dedicated plug-ins (e.g., ns-3) and validation in simulation of the overall nested framework; deployment and validation on real vehicles.

- The robustness of the proposed distributed convoy control algorithm allows for mixing manual and autonomous cars.

Together with the flexibility feature of the designed algorithm, the robustness of the distributed approach compensates the manual vehicle inaccuracy in terms of actuation and integrates these vehicles within convoys of automated cars. This enables us to address one of the goals of the project, the transition from highway solely populated by manual cars to a fully automated car market.

- Kinetic-based distributed control algorithms can be used on vehicles having different dynamics.

Dynamics differences can be compensated by adding dynamics constraints on top of the algorithm natively designed using exclusively kinematic considerations.

7.1.1.2 Hierarchical Semi-decentralized Control

Cooperative control is highly dependent on the perception and communication components. However, since the researches on four axes proceeded in parallel, we do not have fully integrated perception and communication models in the cooperative control development. This implies that the initial assumptions on these two components are crucial for the successful integration at the end. An experience from this project is that both perception and communication should be ready when we develop control algorithms, or initial assumptions should be as close as possible to the reality.

A second lesson of AutoNet2030 is the possibility to develop new algorithms for cooperative control. The fully decentralized approach was known in robotics and has been adapted to the automotive context. Hierarchical strategies were also known, but AutoNet2030 demonstrated with the algorithms of Armines that they can meet the requirements of ITS applications. Mixed techniques have the potential to combine the best of distributed approaches (scalability, reactivity, local view) and centralized approaches (efficiency, coordination, global view).

7.1.2 Perception Layer to Cope with Different Vehicle Platforms

As already briefly mentioned in section 4.2 the AutoNet2030 perception was basically implemented for two different middleware instances during the project lifetime. At the beginning of the project this was a vADASdeveloper-based Windows application while at the end it was changed to a more embedded-compatible version without any direct GUI support which runs under Linux. In fact, this transition generated some additional effort. For future projects it might be a better option to directly start with the embedded version of the perception.

As the perception is a kind of facility layer which connects the hardware sensors towards the application, it is naturally rather sensitive to modifications (either planned or not) of the underlying interfaces and implementations. For example, the temporary outage of a sensor due to a broken physical connector may lead to an undetected degradation of the performance of the perception. As the perception is implemented as a probabilistic system, this unexpected behavior is not always directly observable and might be partly compensated by other sensors. For a complex system, such as an automated vehicle more additional plausibility and consistency checks should be implemented in parallel to the perception in order to handle complexity and safe detection of potential failure states of the whole system.

7.1.3 Building High-accuracy Positioning Solutions

Having a highly accurate and reliable vehicle positioning technology is of prime importance for the reliable execution of automated driving maneuvers. It has been found that without accurate positioning, the uncertainty of local dynamic map increases, and it becomes more difficult to maintain oscillation-free formation or to execute maneuvers reliably. The main requirement on the positioning accuracy is to be able to distinguish reliably the current lane, so that GNSS positioning may work as a complementary and redundant sensor to camera based lane sensing.

After testing different technologies, the project has settled on integrating a positioning solution consisting of two components:

- The cooperative broadcasting of RTK correction data for positioning accuracy improvement, implemented in local proximity RSUs (provided by BroadBit)
- An innovative and highly accurate GNSS positioning algorithm (provided by FlowScape, who have been subcontracted by Scania)

In the field tests it has been found that resulting positioning technology had less than +/- 10 cm accuracy deviations even during high-speed driving. Figure 19 illustrates the logged accuracy of our implementation. The left part shows the radar tracking of the vehicle in front of the ego-car, while the right side shows its movement trace obtained from its communicated position information (brown line). As can be seen, the correct driving lane is always unambiguously identified. In conclusion, the implemented positioning technology is considered to be very well matching the positioning accuracy requirements of automated driving. Future efforts may include addressing issues related to overhead obstacles or urban canyons.

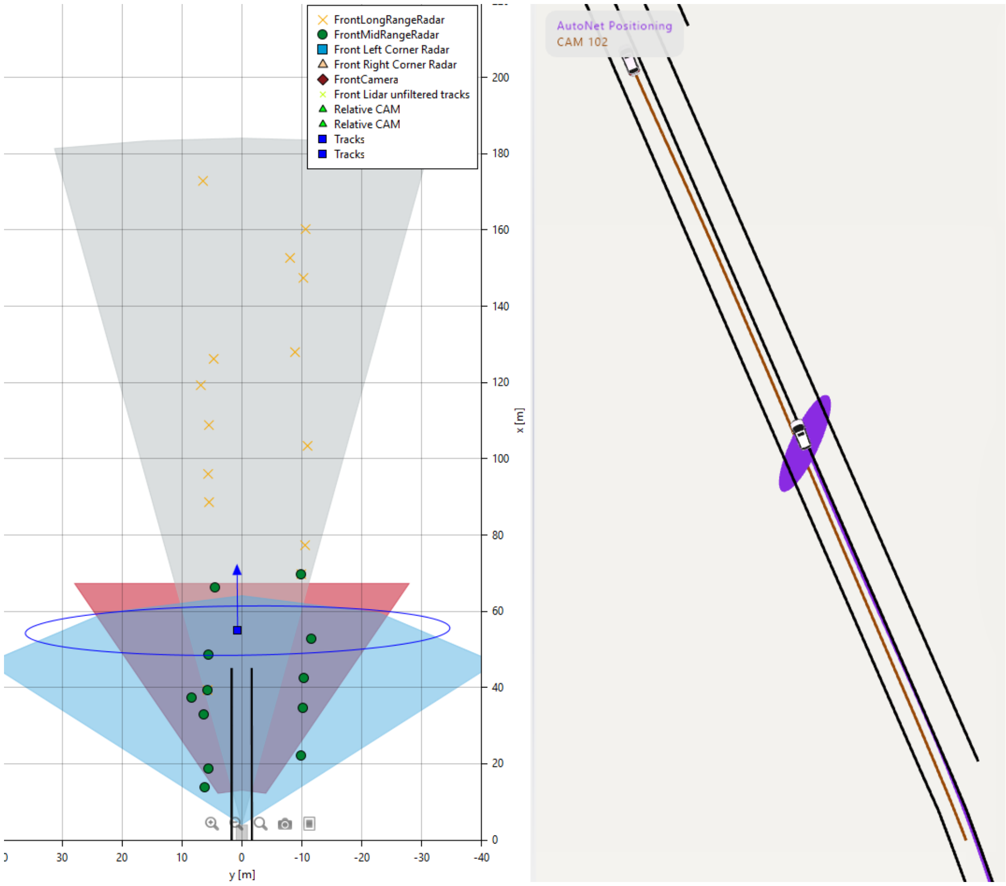


Figure 19 – Illustration of the achievable accuracy during experimentation.

7.1.4 V2X Communications for Cooperative Automation

The V2X communication system used in the AutoNet2030 project is based on an existing implementation that was developed in previous projects on cooperative ITS for the first generation of V2X (1G-V2X, see Section **Error! Reference source not found.**). During the project, the implementation was considerably enhanced to support the functional requirements for cooperative automated driving. The evolution from a stable and well-tested communication system to an experimental system enabled the project partners to focus on the core aspects of the AutoNet2030 project, *i.e.*, on the functional enhancements for cooperative automated system. Instead of developing a full communication system, this evolutionary development process facilitated the integration and testing of the sub-systems perception, HMI and maneuver controller with the V2X sub-system. As a result, the project was able to pursue the full cycle of development from design and specification to implementation and experimental validation.

Furthermore, the R&D efforts for V2X communication were closely linked with the ongoing standardization for V2X communication (see Section **Error! Reference source not found.**). On one side, the project was able to achieve a high impact on the design of future V2X communication. On the other side, the project partner had an advantage because the developed system can be regarded as future-proof.

Another aspect of V2X communication is related to the fact that the AutoNet2030 has carried experiments with a highly experimental system, which explored new approaches not only in V2X communication, but also in the area of perception, distributed algorithms for maneuver control and motion planning as well as HMI. The overall system is highly complex and requires a tremendous amount of efforts for repeating cycles of specification, implementation and testing. In such a process, the use of tools for monitoring of experiments by means of V2X communication is highly beneficial. It allows for evaluation and analysis of experiments, both remote/real-time as well as offline. Such tools would need to be developed at an early stage of the project to be available when experiments are carried out.

7.1.5 Building HMI Applications for Connected Vehicles

During the AutoNet2030 project we tried a first HUD solution that allow to project the designed wireframe images using a projector and a glass combiner placed on the car dashboard in front of the driver. With this first solution we noticed that the field of view, though respecting the specifications of the product, turned out to be quite narrow and could not display in a good way all the displayed contents for the project purpose. We then changed to an alternative HUD based on an Android device (such as the secondary device solution) but this time placed horizontally just underneath the windshield projecting up. In the area of the windscreen above it two optical reflecting films were placed, to obtain the head up display. This solution proved to be effective to display the contents. It also had other advantages for the prototype: it was more stable in terms of connectivity and also in the software module development (Android App). Moreover, after preliminary testing, a common HMI design principle was confirmed: text message projection should be avoided on the HUD. This holds regardless the HUD solution as such messages tend to distract the driver from focusing on the road.

The AutoNet2030 HMI prototype with the two specified visualizers has been tested both in the laboratory and in the test-track where AutoNet2030 use-cases were tested in real driving scenarios. During the first testing phase we tried to use LCM messages for all interfaces of the HMI system. However, we were experiencing a significant packet loss over the connection between the HMI system and the HMI devices, since the UDP multicast - which is the transport mechanism used by the LCM - is often unreliable over WiFi. To achieve the desired data flow rate, it was decided to connect each device to the system using a TCP connection.

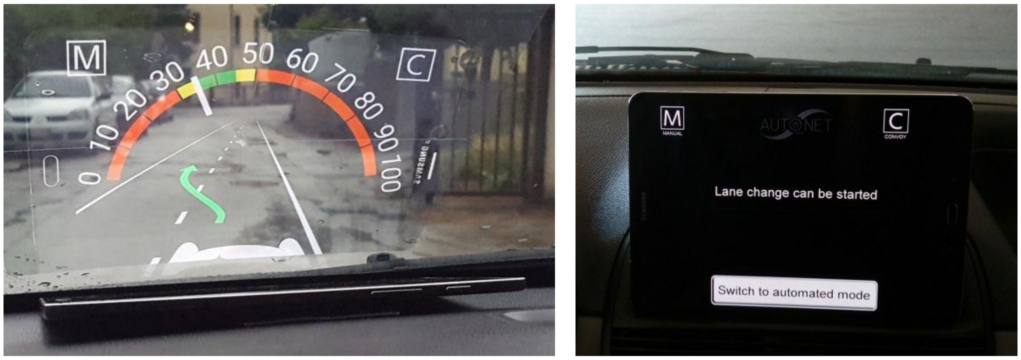


Figure 20 AutoNet2030 HUD (left) and secondary (right) realistic projections

The results of our tests showed that the HMI modules successfully facilitate the establishment of a communication channel and reach the final Android devices. At the same time our results gave positive feedback on the applicability of an Android device for HMI purposes (Figure 20). Finally, a HMI library was developed and provided in order not to lose the advantages of the LCM usage in the development and testing phase (decoupling of modules, efficient cooperation between partners etc., see Section 7.1.6).

7.1.6 Integrated Systems for Future Vehicle Automation

The system integration in a project such as AutoNet2030, where several partners contribute different modules to a complete system, is always a challenge. Within AutoNet2030 a tool-chain was used involving LCM [**3**] for middleware and decoupling of modules, and Docker [**2**] for isolation and orchestration of modules. This proved to be especially successful as the AutoNet2030 team was really able to work independently from each other and in the end a very stable system integration was reached.

- One thing to note is that the possible work decoupling will put harder requirements on defining interfaces and integration points. This to ensure a stability in between the modules and that the different partners are using and providing data of the system in the correct way. To ensure this, projects employing the AutoNet2030 way of working with standardized interfaces and isolated modules, must focus on architectural questions and specifications of the interfaces very early in the project. Furthermore, a product owner must be in place that has the final responsibility of that different modules are working and interacting in the correct way.

8 References

[1] M. Obst, A. Marjovi, M. Vasic, I. Navarro, A. Martinoli, A Amditis, P. Pantazopoulos, I. Llatser, A. de La Fortelle, X. Qian, "Challenges for Automated Cooperative Driving - The AutoNet2030 Approach," in *Automated Driving*.: Springer, 2016.

[2] J. Turnbull, *The Docker Book: Containerization is the new virtualization*.: http://www.dockerbook.com, 2014.

[3] S. A. Huang, E. Olson, and D.C. Moore, "LCM: Lightweight Communications and Marshalling," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2010, pp. 4057-4062.

[4] M. Mesbahi and M Egerstedt, *Graph theoretic methods in multiagent networks*.: Princeton University Press, 2010.

[5] S. Gowal, R Falconi, and A. Martinoli, "Local graph-based distributed control for safe highway platooning," in *IEEE International Conference on Intelligent Robots and Systems*, 2010.

[6] R. Falconi, S. Gowal, and A. Martinoli, "Graph based distributed control of non-holonomic vehicles endowed with local positioning information engaged in escoring missions," in *IEEE International Conference on Robotics and Automation*, 2010.

[7] A Marjovi, M Vasic, J. Lemaitre, and A. Martinoli, "Distributed graph-based convoy control for networked intelligent vehicles," in *IEEE Intelligent Vehicles Symposium*, Seoul, 2015, pp. 64-70.

[8] R.C. Arkin, "Motor schema-based mobile robot navigation," *The International Journal of Robotics Research*, vol. 8, no. 4, pp. 92-112, 1989.

[9] R.C Eberhart and J. Kennedy, "A new optimizer using particle swarm theory," in *6th International Symposium On Micro Machine & Human Science*, 1995.

[10] Webots. Webots. [Online]. https://www.cyberbotics.com/

[11] RO2IVSim. [Online]. http://disal.epfl.ch/RO2IVSim

[12] S. Gowal, Y Zhang, and A. Martinoli, "A realistic simulator for the design and evaluation of intelligent vehicles," in *13th International IEEE Intelligent Transportation Systems Conference*, 2010, pp. 1039-1044.

[13] I. Navarro, F. Zimmermann, M. Vasic, and A. Martinoli, "Distributed Graph-based Control of Convoys of Heterogeneous Vehicles using Curvilinear Road Coordinates ," in *IEEE Intelligent Transportation Systems Conference*, 2016.

[14] X. Qian, I. Navarro, A. de La Fortelle, and F. Motarde, "Motion Planning for Urban Autonomous Driving using Bézier Curves and MPC," in *IEEE Conference on Intelligent Transportation Systems (ITSC)*, 2016.

[15] X. Qian, A. de La Fortelle, and F. Moutarde, "A hierarchical model predictive control framework for on-road formation control of autonomous vehicles," in *IEEE Intelligent Vehicles Symposium*, 2016.

[16] X. Qian, J. Gregoire, A. de La Fortelle, and F. Moutarde, "Priority-based coordination of autonomous and legacy vehicles at intersections," in *17th IEEE International Conference on Intelligent Trasportation Systems (ITSC)*, 2014.

[17] X. Qian, F. Altché, A. de La Fortelle, and F. Moutarde, "A Distributed Model Predictive Control Framework for Road-Following Formation Control of Car-like Vehicles," in *14th International Conference on Control, Automation, Robotics and Vision, ICARCV*, 2016.

[18] L. Hobert, A. Festag, I. Llatser, L. Altomare, and A. Kovacs, "Enhancements of V2X Communication in Support of Cooperative Autonomous Driving," *IEEE Communications Magazine*, vol. 53, no. 12, pp. 64-70, December 2015.

[19] [ETSI] ETSI EN 302 636-4-1, "Intelligent Transport Systems (ITS); Vehicular Communications; GeoNetworking; Part 4: Geographical addressing and forwarding for point-to-point and point-to-multipoint communications; Sub-part 1: Media-Independent Functionality,".

[20] I. Llatser, S. Kühlmorgen, A. Festag, and G. Fettweis, "Greedy Algorithms for Information Dissemination within Groups of Autonomous Vehicles," in *IEEE Intelligent Vehicles Symposium*, Seoul, 2015.

[21] P. Pantazopoulos, L. Altomare, K. Pagle, A.Toffetti, P. Lytrivis and A. Amditis, "Designing a Dual-display HMI System for Connected Vehicles: the AutoNet2030 Approach," in *ITS World Congress*, Melbourne, Australia, 2016.

9 Annex A: Abbreviations and Terminology

Term	Definition
1G-V2X	First Generation of V2X Communications
ADAS	Advanced Driver Assistance System
ADASIS	Advanced Driver Assistance Systems Interface Specifications
APD	Absolute Position Differencing
CAD	Connected Automated Driving
CAM	Cooperative Awareness Message
CABS	Cooperative Awareness Basic Service
CCCS	Convoy Control Communication Service
CLCS	Cooperative Lane Change Service
CTRA	Constant Turn Rate and Acceleration
CSS	Cooperative Sensing Service.
CV	Constant Velocity
ECU	Electronic Control Unit
ETSI	European Telecommunications Standards Institute
GNSS	Global Navigation Satellite System
GNSS-RTK	Global Navigation Satellite System – Real Time Kinematic (High precision *i.e.*, cm-level positioning using pseudo ranges from satellite based geo networking and fixed reference base stations)
FOV	Field Of View (Surveillance area of perception sensor)
GBF	Greedy Broadcast Forwarding
GF	Greedy Forwarding
GMF	Greedy Multicast Forwarding
DENM	Decentralized Environmental Notification Message

HMI	Human machine Interface
ITS	Intelligent Transport System
LCM	Lightweight Communications and Marshalling
LDM	Local Dynamic Map
LDMLite	Local Dynamic Map Lite
MAC	Media Access Control
MOT	Multiple Object Tracking
MPC	Model Predictive Control
MPP	Most Probable Path
N&T	Networking & Transport
NCR	Node Coverage Ratio
NS-3	Network Simulator 3
ODE	Open Dynamic Engine
OOSM	Out Of Sequence Measurement
R&D	Research and Development
RBTP	Reliable Basic Transport Protocol
RTK	Real Time Kinematics.
SAE	Society of Automotive Engineers
SoA	State-of-the-art
UTM	Universal Transverse Mercator
VANET	Vehicular Ad-hoc Network
V2V	Vehicle-to-Vehicle
V2X	Vehicle-to-X