University of
South-Eastern Norway

www.usn.no

FMH606 Master thesis

Industrial IT and Automation

Warning drivers about vehicle wrong way ahead over Nordic Way interchange and cellular network

Alexander Svindseth

Faculty of Technology, Natural sciences and Maritime Sciences
Campus Porsgrunn

University of
South-Eastern Norway

www.usn.no

Course: FMH606 Master thesis 2021

Title: Warning drivers about vehicle wrong way ahead over Nordic Way interchange and cellular network

Pages: 65

Keywords: C-ITS, Wrong way driver, 5G, LTE, Nordic Way Interchange

Student: Alexander Svindseth

Supervisor: Hans-Petter Halvorsen

Tomas Levin (Statens Vegvesen)

External Partner Statens vegvesen (Norwegian Public Road Administration)

Summary:

Meeting vehicles in wrong direction on a divided highway is a dangerous situation that might cause a high energy collision. If one can warn other drivers that a vehicle is heading in their direction using C-ITS, this will help the drivers to avoid the situation. Testing if this can be possible with today's available technology with existing cellular network and the Nordic Way interchange architecture is therefor of interest. To test this, a simplified C-ITS system has been developed and tested on a divided highway to gather important metrics for warning drivers about wrong way drivers. The results show the possibility that vehicles can report their position to a backend service, that can detect if a vehicle drives wrong way and publish this to an interchange node and finally, other backend services can then publish a warning to other vehicles within a time delay that is low enough for other drivers to react.

The University of South-Eastern Norway takes no responsibility for the results and conclusions in this student report.

Preface

This master thesis marks the end of four years of studies as a part time student at the University of South-Eastern Norway, where most of the studies have been done besides fulltime job. The first 1,5 years as an employee at Siemens Oil & Gas and the latter in Norwegian Public Road Administration. The motivation for starting at the studies was to acquire knowledge making me better suited for the job I was doing for Siemens, mostly working with process control and safety systems in the oil and gas industry. I was motivated by gaining a deeper understanding for how to control and optimize a process for increasing production in a safe way. As I later changed to work for the NPRA, the interest shifted to automated vehicles and their interaction with roads, understanding that the goal is still the same; optimize the traffic in a safe way.

With delivery of this thesis, there is of course many to thank. My supervisors, Tomas Levin, and Hans-Petter Halvorsen for their supervision during the writing of this thesis.

Thanks to Ørjan Tveit, for taking the time to read through my thesis and giving me valuable feedback during his Christmas holyday.

Thanks to Christian Berg Skjetne for his support with development of the code used in the test system for this thesis and giving details and access to the Nordic Way interchange.

A special thanks goes to my dearly beloved wife, who has stayed with me through these years as a part time student. When I started the studies, we were boyfriend and girlfriend, while we now are married and have our own family, that puts the years into another perspective. It takes sacrifice from both when one is a part time student for many years.

Bergen, 17.01.2022

Contents

Preface 3

Contents 4

List of figures 6

List of tables 8

Nomenclature 9

1 Introduction 11

1.1 Research objective 12
1.2 Methods 12
1.3 Scope 12
1.4 Definitions 13
1.5 Report structure 13

2 ITS communication, messages, and architecture 14

2.1 ITS-station communication architecture 14
2.2 Messages 16
2.2.1 Message headers 16
2.2.2 Cooperative Awareness service and messages 17
2.2.3 Decentralized Environmental Notification 18
2.3 Nordic Way interchange 19
2.4 Cellular network 21

3 Use cases and services 24

3.1 Knappe tunnel 24
3.1.1 Real life wrong way driver episode 25
3.2 C-roads - Hazardous Location Notification - Alert Wrong Way Driving 29
3.3 ETSI Wrong way driving use cases 29
3.4 Other related work 31

4 Test and test system 32

4.1 Cloud system 32
4.2 Onboard unit 33
4.2.1 Physical setup of onboard unit 35
4.2.2 Mikrotik 5G router 36
4.2.3 Time synchronization with GNSS as source 36
4.2.4 Vehicle positioning 37
4.2.5 tshark 38
4.3 Backend service 39
4.4 Field trial 40
4.4.1 Test parameters 40
4.4.2 Test area 40
4.4.3 Performing the test 41

5 Analysis of data from field trial 43

5.1 Tools used for analyzing the data 43
5.2 Collected data 43

5.3 Latencies in the system 44
5.4 Correlation with signals from cellular network 48
5.5 Outliers 51
5.6 Driving route 51

6 Discussion 54

6.1 Use cases 54
6.2 Latency 54
6.3 Cellular 55
6.4 Positioning 56
6.5 Future work 56

7 Conclusion 57

8 References 58

9 Appendices 64

List of figures

Figure 1-1 H2 - Norwegian national main road, Handbook N100 revision 2021, an example of a divided carriageway [6] 13

Figure 2-1 Nordic Way interchange ecosystem visualizing how cars can communicate through backend systems and over an interchange network 14

Figure 2-2 ITS station reference architecture with possible elements[7] 15

Figure 2-3 Two alternatives of secured message structure[10] 17

Figure 2-4 CAM message suggestion for a passenger car 18

Figure 2-5 General structure of a DENM[13] 19

Figure 2-6 Deployment model with Interchange entities and C-ITS actors[14] 21

Figure 2-7 Connectivity options with 5G-NSA and 5G-SA[16] 22

Figure 3-1 Car slowing down on entrance ramp and uses blinkers to signalize it is going to turn left 25

Figure 3-2 Car about to pass the road marking 26

Figure 3-3 Near collision avoided by the emergency breaking of the approaching car 26

Figure 3-4 Collision avoided, and the wrong way car is over in the left most lane. 27

Figure 3-5 Car have entered the zone for the next camera and is using emergency flashers, probably to signalize to others that it is going wrong way. 27

Figure 3-6 Inverse direction is triggered by AID-system 28

Figure 3-7 Warning of other drivers by lowering the speed limit and igniting emergency evacuation lights 28

Figure 3-8 End to end latency requirments for collision avoidance[24] 30

Figure 4-1 Overall test system showing the connection and protocols between the different components in the test system 33

Figure 4-2 Onboard unit sequence diagram 34

Figure 4-3 Onboard unit - system abstraction 34

Figure 4-4 Onboard unit - physical setup showing the connection between Raspberry Pi, two u-blox GNSS-receivers and the Mikrotik 5G router. 36

Figure 4-5 Waveshare MAX-M8Q GNSS-extension board used for time synchronization over GNSS[45] 37

Figure 4-6 Sequence diagram for backend service 39

Figure 4-7 Test performed on E39 including ramps 40

Figure 4-8 5G coverage map[70] 41

Figure 4-9 Placement of GNSS-antennas and cellular network antennas on the car roof 42

Figure 5-1 Measurement points in test system. Green point shows where the measurement is logged and naming of that point. 43

Figure 5-2 Round trip time from MpA to MpF with outliers....45

Figure 5-3 Round trip time from MpA to MpF without outliers....47

Figure 5-4 Round-trip time plotted together the number of CA bands and LTE/5G for dataset A....48

Figure 5-5 Round-trip time plotted together the number of CA bands and LTE/5G for dataset B....48

Figure 5-6 Round-trip time plotted together the number of CA bands and LTE/5G for dataset C....49

Figure 5-7 Correlation matrix between upload and download latency and cellular network signals for dataset A....49

Figure 5-8 Correlation matrix between upload and download latency and cellular network signals for dataset B....50

Figure 5-9 Correlation matrix between upload and download latency and cellular network signals for dataset C....50

Figure 5-10 Time components in outliers for the dataset B and C....51

Figure 5-11 Collected positions from dataset A with different colored dots based on the data-class....52

Figure 5-12 Collected positions from dataset B with different colored dots based on the data-class....53

Figure 5-13 Collected positions from dataset C with different colored dots based on the data-class....53

List of tables

Table 4-1 Start and end time for the test. Time zone is UTC 0 41
Table 5-1 Number of collected data related to the measurement points in Figure 5-1............ 44
Table 5-2 Mean value for round-trip, upload, download and “backend-to-backend”-latency 46

Nomenclature

3GPP	3rd Generation Partnership Project
4G	Fourth generation cellular network technology
5G	Fifth generation cellular network technology
5G-NSA	5G-Non standalone
5G-SA	5G-Standalone
AID	Automatic Incident Detection
AMQP	Advanced Message Queue Protocol
API	Application Programming Interface
ASN.1	Abstract Syntax Notation One
AWWD	Alert Wrong Way Driving
BI	Basic Interface
BTP	Basic Transport Protocol
C-ITS	Cooperative Intelligent Transport System
CA	Cooperative Awareness
CAM	Cooperative Awareness Message
CCAM	Cooperative, Connected and Automated Mobility
CEP	Circular Error Probable
CoAP	Constrained Application Protocol
DATEX	Data Exchange
DEN	Decentralized Environmental Notification
DENM	Decentralized Environmental Notification Message
ETSI	European Telecommunications Standards Institute
EPC	Evolved Packet Core, the core network in LTE
GN	GeoNetworking
GNSS	Global navigation satellite system
GPIO	General Purpose Input/Output
GPS	Global Positioning System
GUI	Graphical User Interface
HLN	Hazardous Location Notification
HSM	Hardware Security Module
I2V	Infrastructure to Vehicle
IDE	Integrated Development Enviroment
II	Improved Interface
IMU	Inertial Measurement Unit
IP	Internet Protocol
ITS	Intelligent Transport System

IVIM	Infrastructure to Vehicle Information Message
LDM	Local Dynamic Map
LTE	Long-Term Evolution
MQTT	Message Queuing Telemetry Transport
NPRA	Norwegian Public Road Administration (in Norwegian: Statens vegvesen)
NTRIP	Networked Transport of RTCM via Internet Protocol
NTP	Network Time Protocol
OBU	Onboard Unit
OEM	Original Equipment Manufacturer
OS	Operating System
OSI	Open System Interconnection
PER	Packed Encoding Rules
PPS	Pulse per second
RAM	Random Access Memory
REST	Representational state transfer
RHS	Road Hazard Signaling
RSSI	Reference Signal Strength Indicator
RSRP	Reference Signal Received Power
RSRQ	Reference Signal Received Quality
SINR	Signal to Interference & Noise Ratio
SIM	Subscriber Identification Module
SRTI	Safety Related Traffic Information
SSH	Secure Shell
TCP	Transmission Control Protocol
TLS	Transport Layer Security
TMC	Traffic Management Center (In Norwegian: Vegtrafikksentral)
UDP	User Datagram Protocol
UE	User Equipment
UML	Unified Modelling Language

1 Introduction

The field Intelligent Transport System is a large field that is in constant evolution. Among the first ITS-system, were the traffic lights that made it possible to control the traffic flow in intersections. Later, ITS-systems as variable speed signs, automatic incident detection, ramp control etc. have been implemented. Common for most of ITS deployed in everyday use, is that the information to the vehicle driver is communicated from the outside of the vehicle. The time is now shifting towards that the driver information is communicated directly in the vehicle and the information is transmitted to the vehicle wirelessly. And in the future; possibly directly to the vehicle instead of to the person in the driver seat.

Within ITS-services, it is usually distinguished between safety-, non-safety- and other services. Non-safety can be Traffic Light Assistance or environmental geofence zones, where the aim is to smoothen traffic flow in traffic light intersections or reduce the pollution within a geographical area. Other services are typically media applications like music, podcast etc.

For the safety-related services the goal is to reduce the risk of accident that harm people and equipment. Example of these services can be slippery road warning, emergency vehicle approaching, stationary vehicle etc.

One of the deadliest types of accidents, are the head-on front-impact collisions, as one, and potentially both, vehicles travel at relative high speed before the impact occurs. On highways that have divided road lanes, traffic volume and speed limits are generally higher as all the vehicles are travelling in the same direction, and the risk of front-impact collision is reduced. Still, it happens that vehicles comes in wrong direction, either by driving in on an exit ramp or stopping and turning on traffic. This phenomenon is denoted a “Wrong Way driver” and causes great risk of high energy collision.

An example from USA [1], we have that during an 11 year period in Arizona, there was 91 fatalities in 245 accidents with wrong way drivers and 65% of the collisions was caused by impaired drivers.

In the public statistics from NPRA[2], it was 14 fatalities in 164 accidents in front collisions on divided roads from 1990 to 2020.

The use case with “Wrong Way driver” is therefore chosen as one of many safety-related use cases as a basis for this project, as it holds many aspects that are important within safety related C-ITS; positioning, reliability, availability, and latency. An important boundary with this use case is that it is about warning other drivers then the one driving wrong way, as one can assume that the wrong way driver have had several chances to detect the driving direction, either by looking at the traffic direction, reading signs or road marking.

This use case is also implemented several places with infrastructure-based systems that relies on sensors installed in or near the road and reports incidents automatically in various ways. Sensors for such a system can be radars, cameras, and inductive loops. Such a system is commonly denoted Automatic Incident Detection (AID), where the “Wrong Way driver” is one of the use cases supported.

There are also exists mechanical systems that punctures the tires of vehicles if it drives wrong way on an exit ramps[3]. These systems are costly to install in the road and can be challenging to operate during Nordic winters. It also doesn’t detect and act on vehicles that stops on the highway and turn on the traffic direction.

This report focuses on how communication technologies can be used to mitigate the problem of wrong-way drivers. The results will give the traffic engineers a good understanding of the performance of new C-ITS systems that have to bring information into the vehicle.

1.1 Research objective

Is it possible to take a leap from existing systems to a system based on C-ITS, with the Nordic Way interchange architecture and communicating over cellular network to warn drivers about wrong way drivers?

1.2 Methods

Perform a literature study on established use cases and services for a definition of how a wrong way ahead system should operate and find the requirements for such a system based on existing C-ITS standards.

A core part of this work is to build a test system and perform field trials to collect metrics. The collected metrics will be analyzed to find the key performance metrics of the system to find out if it is possible to fulfill the requirements for the selected use case.

1.3 Scope

In this project, a full service of the use case will not be implemented, but rather collecting important metrics for analyzing how a service can perform. The software stack for an ITS-station is advanced software that needs to follow many standards, and it would probably take many years of full-time development to make a complete software stack. Hence extracting the critical parts of the service was done, focusing on the use of an interchange and use of the cellular network.

Personnel travelling in a vehicle are potentially identifiable by e.g., license number plates or movement pattern. Person identifiable information are regulated by the general data protection regulation (GDPR). When the same vehicle is connected through cellular communication in almost the same manner as a phone, it becomes possible to track the vehicles movement and thus the personnel travelling with it as well. This project does not take aim to solve issues related to GDPR, as the result is the performance of core components of a service and not the public service.

The test is to be performed under naturalistic conditions but avoiding “corner-cases” that are known to impact the metrics gathered. In practical terms, the test will try to avoid areas with known for poor GNSS and cellular network coverage.

Much research and standardization have been performed around the use of short-range communication for C-ITS purposes, it isn’t readily available as very little have been deployed at large scale. This report then focuses on the use of wide-range communication, and then specifically over cellular network.

No human machine interface (HMI) will be installed in the test system as it is hard to measure the reaction time for when the information will be present on an HMI. For safety reasons such interactions should be tested in simulator studies.

1.4 Definitions

The basic definition for “Wrong way driver” is taken from the SRTI Delegated regulation 886/2013[4]

> *‘wrong-way driver’ means a vehicle travelling on the wrong side of a divided carriageway against the oncoming traffic;*

According to Cambridge Dictionary a “dual carriageway” is[5]:

> *a road that has an area of land in the middle, dividing the rows of traffic that are moving in opposite directions*

Even though the SRTI-regulation states “divided carriageway”, this is assumed to be the same as “dual carriageway”.

In Norwegian terms, this typically means a highway with four lanes or more that is divided by land and/or barrier. It can also be a two/three lane road divided by barrier as shown in Figure 1-1.

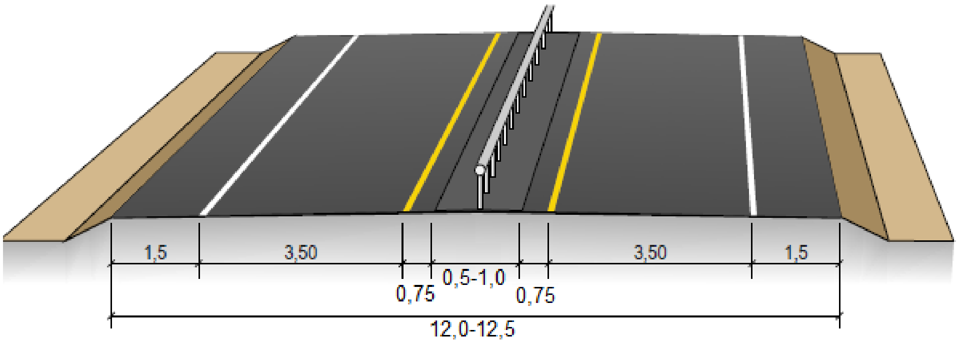


Figure 1-1 H2 - Norwegian national main road, Handbook N100 revision 2021, an example of a divided carriageway [6]

1.5 Report structure

In chapter 2, the general ITS-architecture standardized by ETSI is presented to give an overview of some of the elements within C-ITS. In addition to the standardization of C-ITS, the architecture of the Nordic Way interchange network, which is a network for exchanging C-ITS messages is explained.

In chapter 3, use cases and relevant related work on the “Wrong way driver” use case is described and briefly discussed. Under one of the use cases, a real-life episode from Knappe tunnel is described to provide insight about what factors are important.

In chapter 4, the test system used for testing and the different components in the system is described. The selection of field trial area and parameters are decided, and data are collected.

Chapter 5 analyses the data collected from the field trial and presents the results.

Chapter 6 discusses the result, and how they affect the “Wrong Way ahead” use case, suggests some possible improvements and suggest future work that will be important.

Chapter 7 answers on the research objective and concludes on the main findings in this report

2 ITS communication, messages, and architecture

This chapter is to give an overview of ITS communication and architecture, it is focusing on the existing standards for Europe, mainly published under ETSI and C-Roads, used in ecosystems like the one in Figure 2-1 that visualizes a possibility of how vehicles can communicate.

- Chapter 2.1 gives a brief overview of the ITS-station architecture and how this relates to the OSI-model.
- Chapter 2.2 gives an overview of the message format for ETSI-messages, and then especially for the messages relevant for the “Wrong way driver”-use case
- Chapter 2.3 takes on the Nordic Way interchange with both the background for this network and a bit of technical details.
- Chapter 2.4 is regarding some key elements in cellular networks.

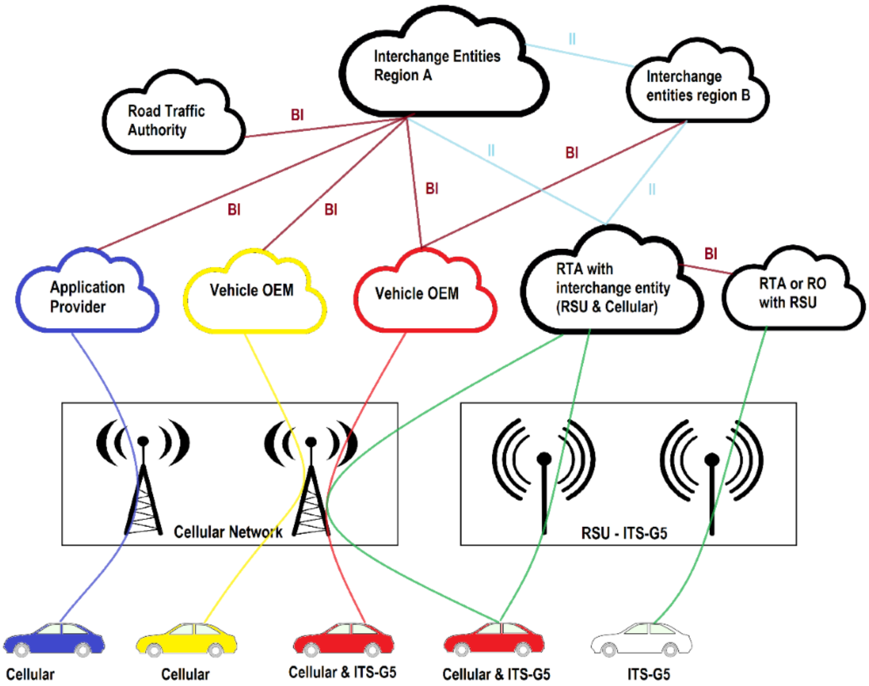


Figure 2-1 Nordic Way interchange ecosystem visualizing how cars can communicate through backend systems and over an interchange network

2.1 ITS-station communication architecture

The ITS-station reference architecture is defined by ETSI EN 302 665[7]. Parts of this architecture are related to the more known Open System Interconnection model (OSI-model). With regards to Figure 2-2, the midsection consists of three layers displayed as grey boxes:

- Access layer: This layer is equal to physical layer and data link layer in the OSI-model. The physical layer can be connected to e.g., cellular network, ITS-G5, ethernet or in-vehicle network.

- Networking & Transport: This layer relates to network and transport layer in the OSI-model and is often utilized by the TCP/IP protocol.
- Facilities: Consists of the session, presentation, and application layer in the OSI-model. The basic services are under this layer, where e.g., DEN and CA services is handled, and generates DENMs and CAMs respectively. This will be further elaborated in chapter 2.2.2 and chapter 2.2.3

For the three remaining boxes in Figure 2-2, they are not in relation to the OSI-model:

- Applications: Here are the ITS-applications grouped into and can probably best be related to business logic in computer science.
- Management: As the name implies, this manages the ITS-station.
- Security: Holds the security functionality for the ITS-station. It includes the Hardware Security Module (HSM), which is a physical device to secure cryptographic keys (e.g., certificates)[8]. In practice this means that every ITS-station need a physical device to be compliant with ETSI EN 302 665.

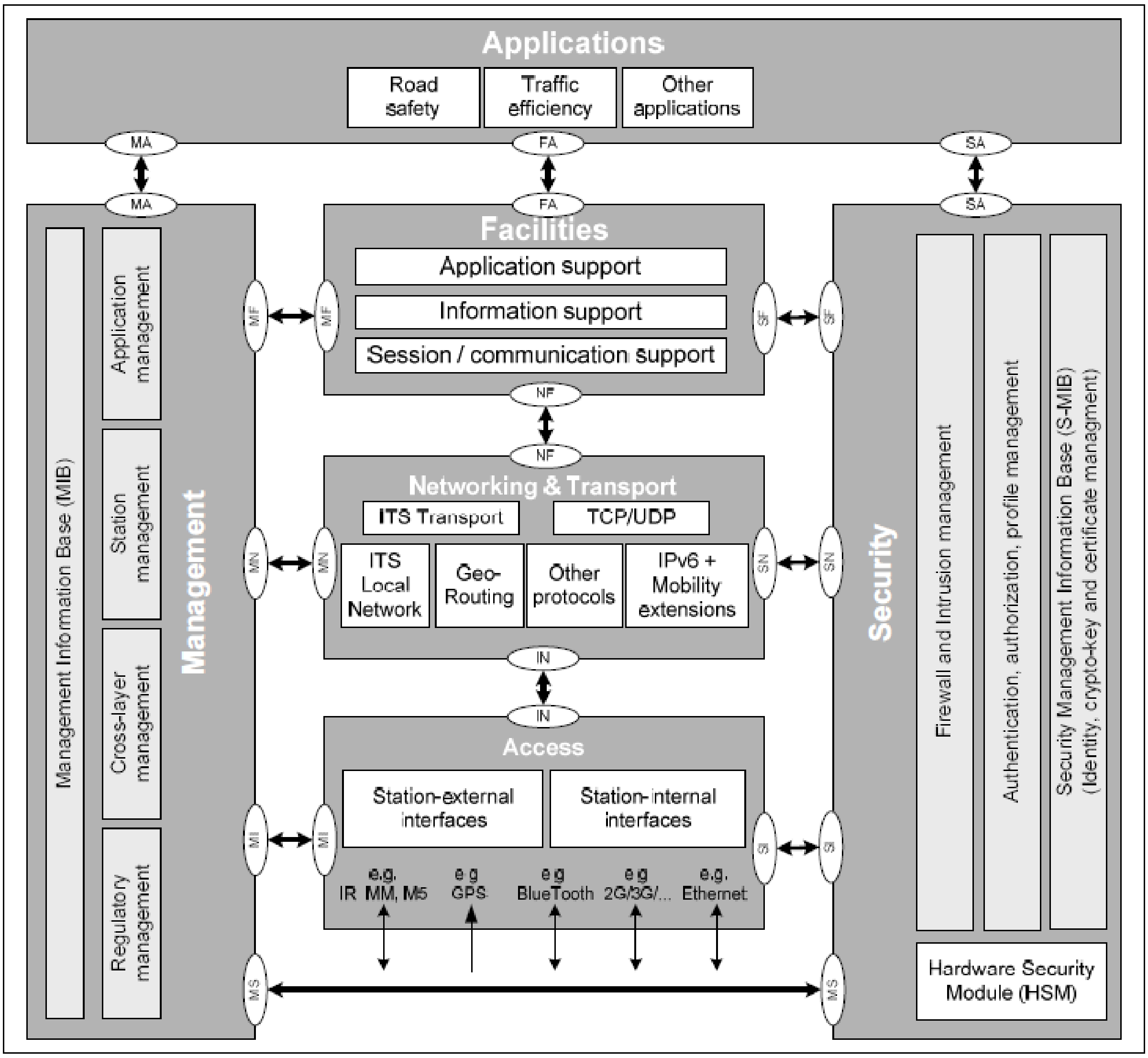


Figure 2-2 ITS station reference architecture with possible elements[7]

2.2 Messages

From the information exchange in Figure 2-1, the communication between the vehicles, or at least between the OEM and service providers, needs to be standardized. There already exists many message standards that have been standardized through ETSI for C-ITS purposes, these messages are commonly denoted as C-ITS-messages. The most relevant messages for this project are the Cooperative Awareness Message (CAM) and Decentralized Environmental Notification Message (DENM) which will be explained further in chapter 2.2.2 and 2.2.3. Other notable messages are among others: SPATEM for traffic lights, MAPEM for road layout and Infrastructure to Vehicle Information Message (IVIM) for in-vehicle information (e.g., speed signs, stop signs etc.)[9]

The messages are packed together with headers in a manner known from most communication protocols and the headers contain information that is common for all messages that are standardized under ETSI. A brief overview of the different header elements and which headers that is relevant to use with cellular communication is reviewed in chapter 2.2.1

2.2.1 Message headers

In Figure 2-3, two message structures are defined. Common for both are the grey and green boxes. The green box is where message generated in the facility layer is encapsulated. The dark grey is the access layer, and the light grey is optional for cellular communication where the message needs to encapsulate in e.g., a TCP/IP-packet.

The upper of the two structures, is the one that is standardized on today. In this the GeoNetworking (GN) basic header, GN common header and Basic Transport Protocol (BTP) headers are mandatory and are added in the networking and transport layer. These headers are essential in short-range communication as the GN targets a specified geographical area, making it possible to drop the message at the networking and transport layer and thus avoid sending it to the facility layer.

The GN common header and BTP is encapsulated by a signature signed with a certificate. This signature is vital for assessing the integrity of the message. To sign this message, the position, timestamp, and certificate is included to the encryption algorithm. From the encryption, the encryption signatures are added to the message header,

If a vehicle is connected to a backend solution over cellular network provided by an OEM or service provider, the vehicle will have a dedicated IP-address and all the information from this backend is routed to this IP.

Since the GN and BTP are first added on the networking and transport layer, this means that the signature also needs to be added on this layer. But for wide range communication, the GeoNetworking doesn't necessarily fill a purpose as the addressing is done on the IP-address. This imposes a problem for message signing since this is done on the network and transport layer and needs to include GN and BTP.

This issue has been addressed in [10], and suggests the lower structure in Figure 2-3. Here it is suggested to sign the message at the facility layer and attaching GN and BTP outside the security envelope as optional headers.

For both structures, additional headers are attached, this can typically be MQTT/TCP/IP-headers or CoAP/UDP/IP-headers. This is visualized by the light grey boxes in Figure 2-3.

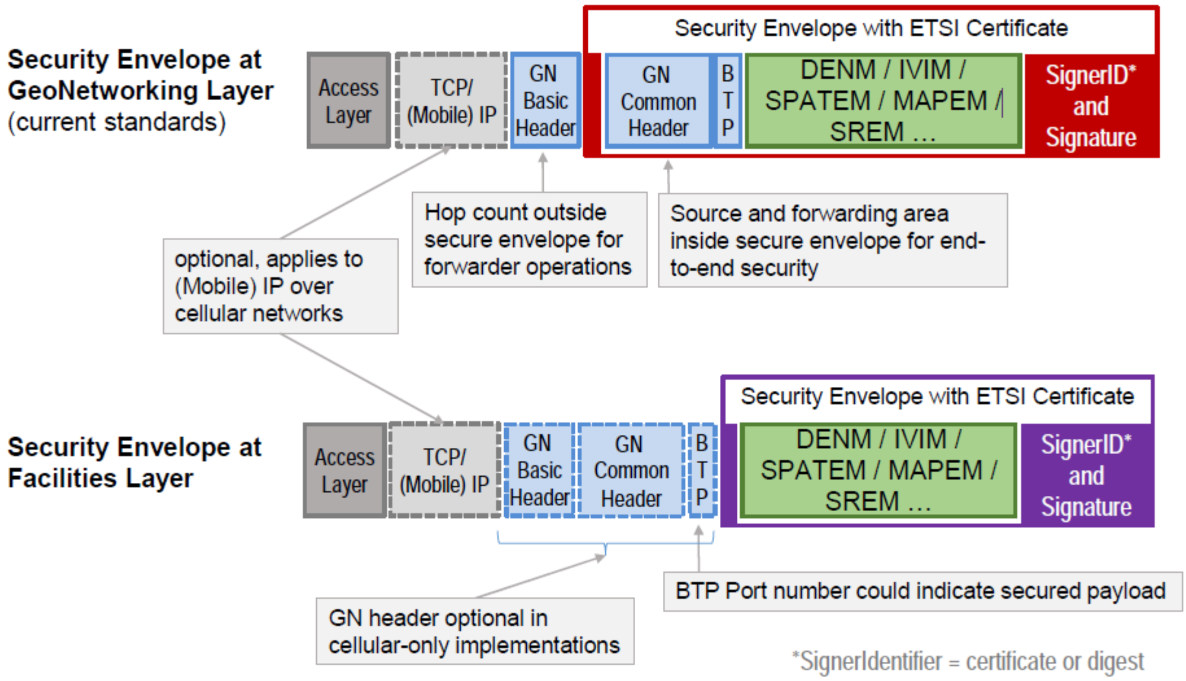

Security Envelope at GeoNetworking Layer (current standards)
Access Layer
TCP/ (Mobile) IP
GN Basic Header
Security Envelope with ETSI Certificate
GN Common Header
BTP
DENM / IVIM / SPATEM / MAPEM / SREM …
SignerID* and Signature
Hop count outside secure envelope for forwarder operations
Source and forwarding area inside secure envelope for end-to-end security
optional, applies to (Mobile) IP over cellular networks
Security Envelope at Facilities Layer
Access Layer
TCP/ (Mobile) IP
GN Basic Header
GN Common Header
BTP
Security Envelope with ETSI Certificate
DENM / IVIM / SPATEM / MAPEM / SREM …
SignerID* and Signature
GN header optional in cellular-only implementations
BTP Port number could indicate secured payload
*SignerIdentifier = certificate or digest

Figure 2-3 Two alternatives of secured message structure[10]

2.2.2 Cooperative Awareness service and messages

Cooperative Awareness service is defined by ETSI EN 302 637-3[11]. As the name implies, this message is for cooperative information between all road users. The aim is to disseminate information that is necessary for the cooperation between C-ITS actors. The Cooperative Awareness service sends out Cooperative Awareness Message (CAM) and can be transmitted between vehicles and roadside units, between vehicles or from physical equipment, e.g., roadblocks. The message shall be transmitted with at least 1 Hz rate and at most a 10 Hz rate.

There are several options in the CAM and typically it will be used on ordinary vehicles, but it has special fields for public transport, emergency vehicles, road works etc. It can also be used for non-moving equipment like roadblocks to emit that a road lane is closed, or road works is ongoing.

Figure 2-4 presents a suggestion in UML-format for a CAM for a passenger car. This figure only shows the most relevant containers for a passenger car, and more containers can be added for special vehicles. Some of the data fields are subject to change for every cycle while driving, like position, altitude, speed. While others are rarely changed, like the vehicle width and stationID

The message content is specified in Annex A in the standard and is described using ASN.1 notation, and Annex B is describing the data elements. For most of the fields, Annex B refers to a common data dictionary, ETSI TS 102 894-2[12], for further specification of the elements.

Upon transmittal or receival of message, it shall be ASN.1 Unaligned PER-encoded and decoded based on an ASN.1 scheme. This reduces the message size substantially.

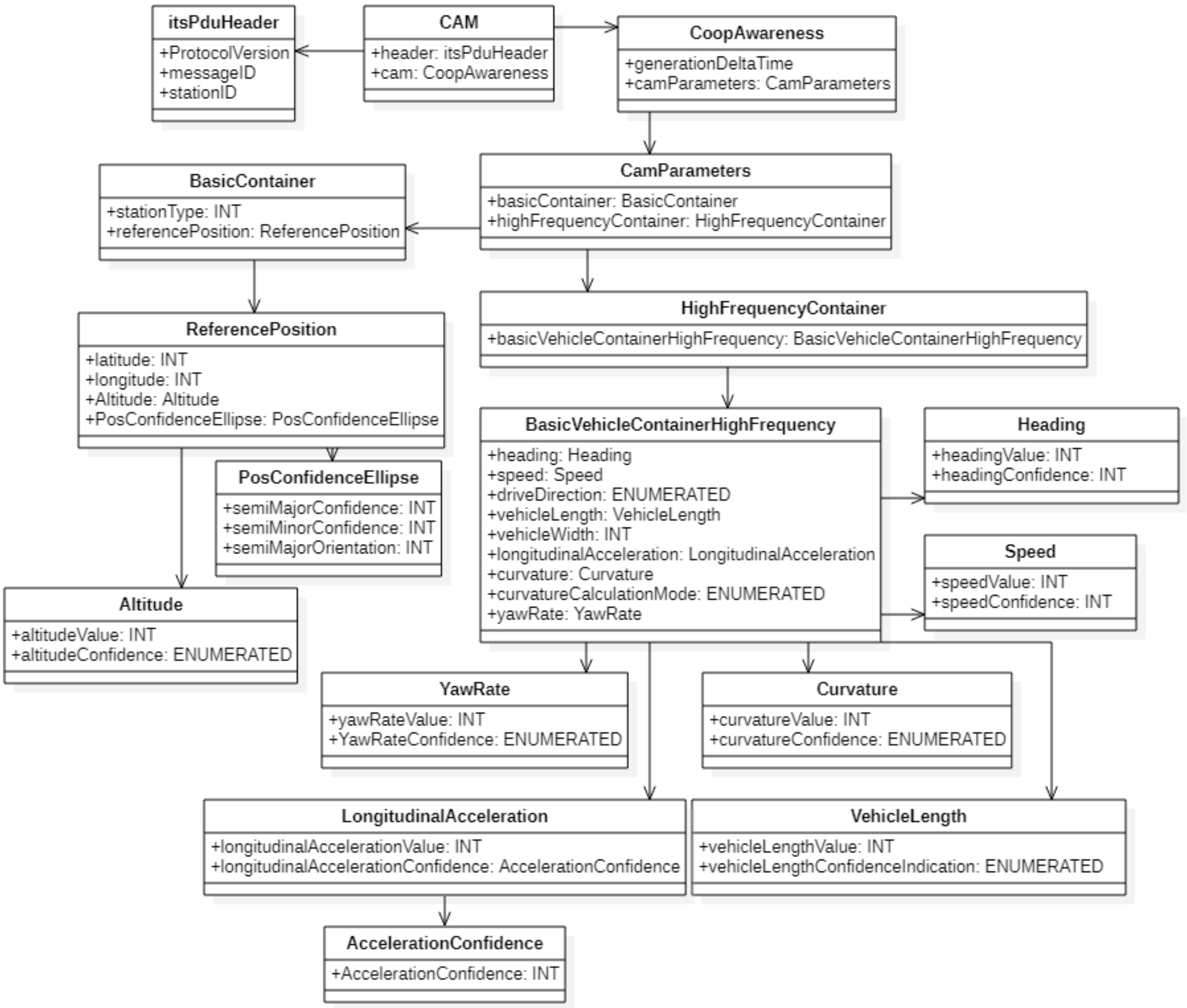

itsPduHeader
+ProtocolVersion
+messageID
+stationID
CAM
+header: itsPduHeader
+cam: CoopAwareness
CoopAwareness
+generationDeltaTime
+camParameters: CamParameters
BasicContainer
+stationType: INT
+referencePosition: ReferencePosition
CamParameters
+basicContainer: BasicContainer
+highFrequencyContainer: HighFrequencyContainer
HighFrequencyContainer
+basicVehicleContainerHighFrequency: BasicVehicleContainerHighFrequency
ReferencePosition
+latitude: INT
+longitude: INT
+Altitude: Altitude
+PosConfidenceEllipse: PosConfidenceEllipse
PosConfidenceEllipse
+semiMajorConfidence: INT
+semiMinorConfidence: INT
+semiMajorOrientation: INT
BasicVehicleContainerHighFrequency
+heading: Heading
+speed: Speed
+driveDirection: ENUMERATED
+vehicleLength: VehicleLength
+vehicleWidth: INT
+longitudinalAcceleration: LongitudinalAcceleration
+curvature: Curvature
+curvatureCalculationMode: ENUMERATED
+yawRate: YawRate
Heading
+headingValue: INT
+headingConfidence: INT
Speed
+speedValue: INT
+speedConfidence: INT
Altitude
+altitudeValue: INT
+altitudeConfidence: ENUMERATED
YawRate
+yawRateValue: INT
+YawRateConfidence: ENUMERATED
Curvature
+curvatureValue: INT
+curvatureConfidence: ENUMERATED
LongitudinalAcceleration
+longitudinalAccelerationValue: INT
+longitudinalAccelerationConfidence: AccelerationConfidence
VehicleLength
+vehicleLengthValue: INT
+vehicleLengthConfidenceIndication: ENUMERATED
AccelerationConfidence
+AccelerationConfidence: INT

Figure 2-4 CAM message suggestion for a passenger car

2.2.3 Decentralized Environmental Notification

DEN is defined in ETSI EN 302 637-2[13], a DENM is typically used to warn vehicles about situations on the road. Opposed to a CAM that is sent periodically at a certain frequency, DENM is event triggered. It is best explained in the standard itself:

> *"A DENM contains information related to an event that has a potential impact on road safety or traffic condition. An event is characterised by an event type, an event position, a detection time and a time duration. These attributes may change over space and over time"*

From Figure 2-5, the DENM message consists of one header and four containers, where three of them is optional.

The ITS PDU Header is the same header for all ETSI-messages, and is defined in [12] where the messageID within the header describes that the message is a DENM.

The management container holds information that is mandatory of all DENM’s, as detectionTime, eventPosition etc.

The situation container hold information about what situation is ongoing for the event notification. It consists of five fields:

- informationQuality: an integer value between 1 to 7 where the lowest value indicates a low quality and the higher value a higher quality.
- causeCode: an integer value that maps to a table in [13] that describes the event,
- subCauseCode: an integer value that gives a more detailed description about the causeCode. E.g., more details about what type of Road Works is ongoing.
- linkedCause: Is an optional field and are the same value as a causeCode, but is used to link several causes together, e.g., stationary vehicle and vehicle breakdown.
- eventHistory: Is an optional field. If the event has happened on several positions, this field shall include the former eventPositions.

A special note should be given to the causeCode and subCauseCode as these have defined codes for Wrong Way driving within the standard.

The À la carte container can hold extra information about the event, for example the lanePostion of the event and impactReduction for a potential collision.

As the name implies, the location container holds the information about where the event is located. This container can hold several traces of where the event is ongoing.

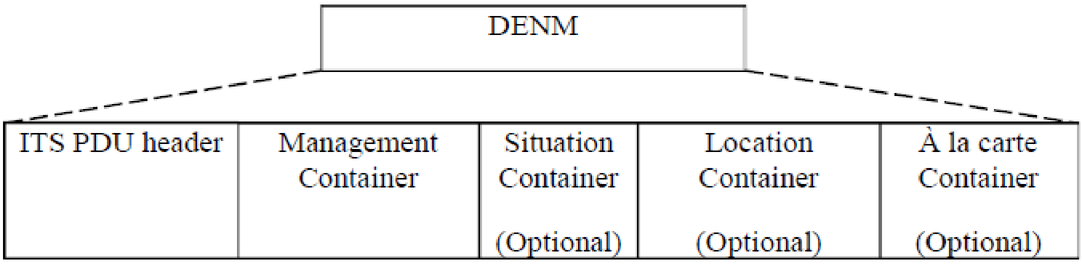

DENM
ITS PDU header
Management Container
Situation Container (Optional)
Location Container (Optional)
À la carte Container (Optional)

Figure 2-5 General structure of a DENM[13]

With the amount of optional information, the size of the DENM can vary depending on the event type and location.

As for the CAM in chapter 2.2.2, the DENM shall be ASN.1 UPER encoded and decoded

2.3 Nordic Way interchange

The Nordic Way ecosystem started with the first Nordic Way project originating from 2015 and have been further developed through Nordic Way 2 and now in Nordic Way 3.
Among the key evolvements through the projects is the development of interchange network that commonly goes under the name Nordic Way Interchange.

In 2015 the cellular communications had come as far, that original equipment manufacturers (OEM) had started connecting vehicles to their backend solutions and providing the drivers information via cellular network. This still limited the OEMs to only speak with similar brand vehicles, and not communicating with vehicles from other OEMs. The idea was therefore to connect all the backends to an interchange network and exchange information in predefined and standardized formats.

This work is also supported by the C-Roads platform in [14] to form a standardization across Europe, and Nordic Way Interchange is aligned with the standard in [14], but in addition, it supports the exchange of DATEX messages.

The interchange consists of two interfaces: Basic Interface (BI) and Improved Interface (II).

The BI is for exchanging messages between interchange nodes and/or service providers or OEMs, denoted as “Interchange Entity” and “C-ITS actor” in Figure 2-6. It allows all actors to publish and subscribe to C-ITS messages. The protocol is AMQP1.0 over TCP/IP secured with TLS1.3.

For all the messages that is to be exchanged over BI, the messages need to be attached with application properties. These properties give information about who the publisher is, what type of service it is, geographical information and message type. For details about the properties, see [14]. All these properties give additional information about the payload. The payload is a C-ITS-message as denoted in chapter 2.2.

Based on the application properties, it is possible to apply filtering on subscribed messages. Typically, a C-ITS actor is only interested in information within a geographical area or information of a certain type. This way, it is possible to filter the messages without reading the payload.

Regarding latency, the requirements state that an interchange shall be able to route a single message within 30ms when the message size is less than 500kB.

The Improved Interface (II) is a control plane interface for the BI and is used between interchanges to share capability of the respective interchanges. It is an optional interface to avoid manual configuration when new interchange nodes are added. Over this interface, each new interchange can post its capabilities to the network of interchanges.

For example, if Trondheim decides to add an interchange that publishes information about all signalized intersections in the city, the interchange will post its capability to the network of interchanges. If a vehicle connected through a different interchange enters Trondheim and want the status on the intersections, it will collect this information from Trondheim’s interchange.

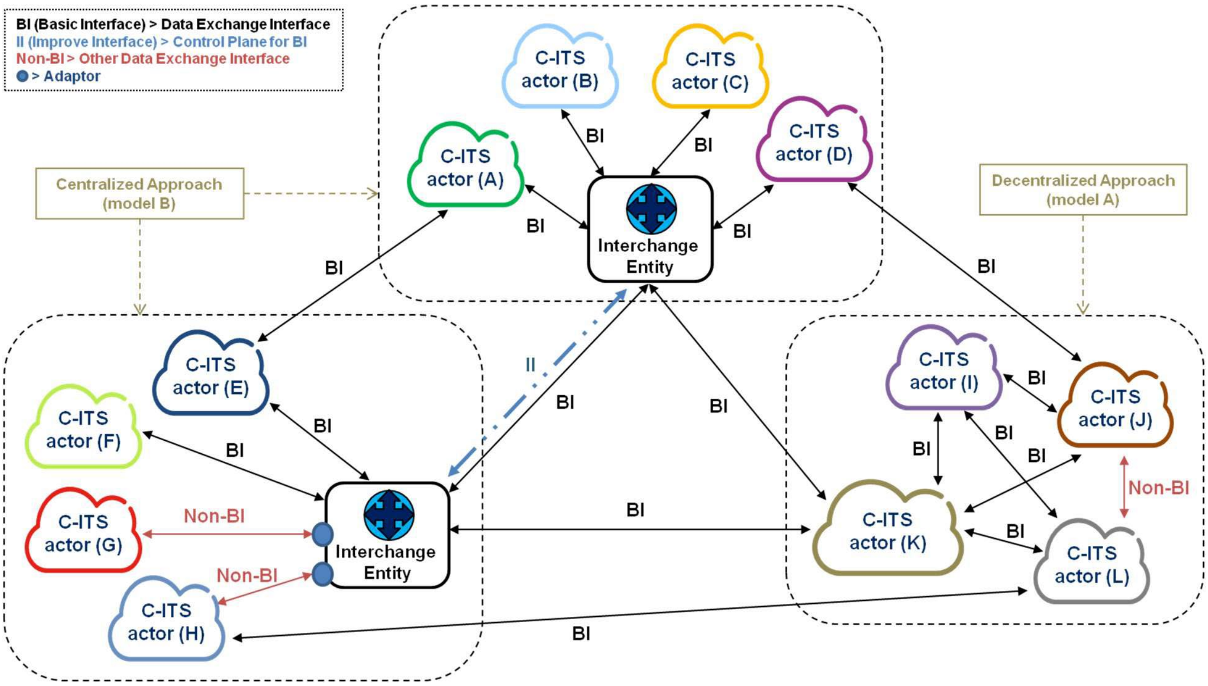


Figure 2-6 Deployment model with Interchange entities and C-ITS actors[14]

Looking at ETSI ITS station reference architecture in chapter 2.1 and comparing it to the interchange architecture in this chapter, it is clear to see that they are unrelated in many ways. This is on purpose, as the interchange doesn't produce or consume any message, opposed to an ITS-station. The interchange only exchange information between actors, and is per definition payload agnostic, as it doesn't add or alter any of the information exchanged.

With the architecture in Figure 2-1 and Figure 2-6, a possibility is that the C-ITS actor can receive vehicle position and a timestamp in a non-ETSI format, and convert this to an ETSI-format, sign it and forward it to an interchange node for other C-ITS actors to pick up. The C-ITS actor then needs to follow the reference architecture in chapter 2.1 and be a centralized ITS-station. As the specification for the interchange requires signed ETSI-messages, and other C-ITS actors will need to get messages in a standardized format that they can verify the origin of. This also means that if it follows the reference architecture, it needs access to a physical HSM, even though the centralized ITS-station runs in a cloud.

2.4 Cellular network

In Norway, there is currently three types of high-capacity cellular network used. LTE (4G)[1], LTE-Advanced (4G+) and 5G-NSA.

The main difference between LTE and LTE-Advanced, is that LTE-Advanced uses carrier aggregation[15], meaning that it can carry data over several frequencies at once and increase the data throughput.

In LTE, the radio protocol uses a control and data plane architecture and is a well-known architecture within networking. Within LTE, the term user plane is used instead of data plane

[1] 4G and 4G+ is used by the Norwegian telecom operators, while LTE and LTE-Advanced is the official names defined by 3GPP

and in this thesis, user plane will be used in conjunction with LTE. Control plane handles signaling and control functions, like authentication of SIM-card. The user plane handles data that is up- and downloaded.

5G-NSA is the 5G-technology that is most widely deployed, giving cellular network operators short time to market by utilizing the existing LTE infrastructure as an enabler for 5G NR. 5G NSA stands for Non-standalone and uses the LTE-network for control signaling[16], meaning that all handling of connection is done in the LTE-network while a 5G-band is added in addition for data carriage with a higher bandwidth than LTE can offer. This is visualized in option 3 in Figure 2-7, where the red line visualizes the control plane data flow, and the black line the user plane. In practical terms, this means that the new features that can be enabled with 5G, like network slicing and ultra-low latency isn't implemented as these rely on 5G Stand Alone (SA) with a 5G core network.

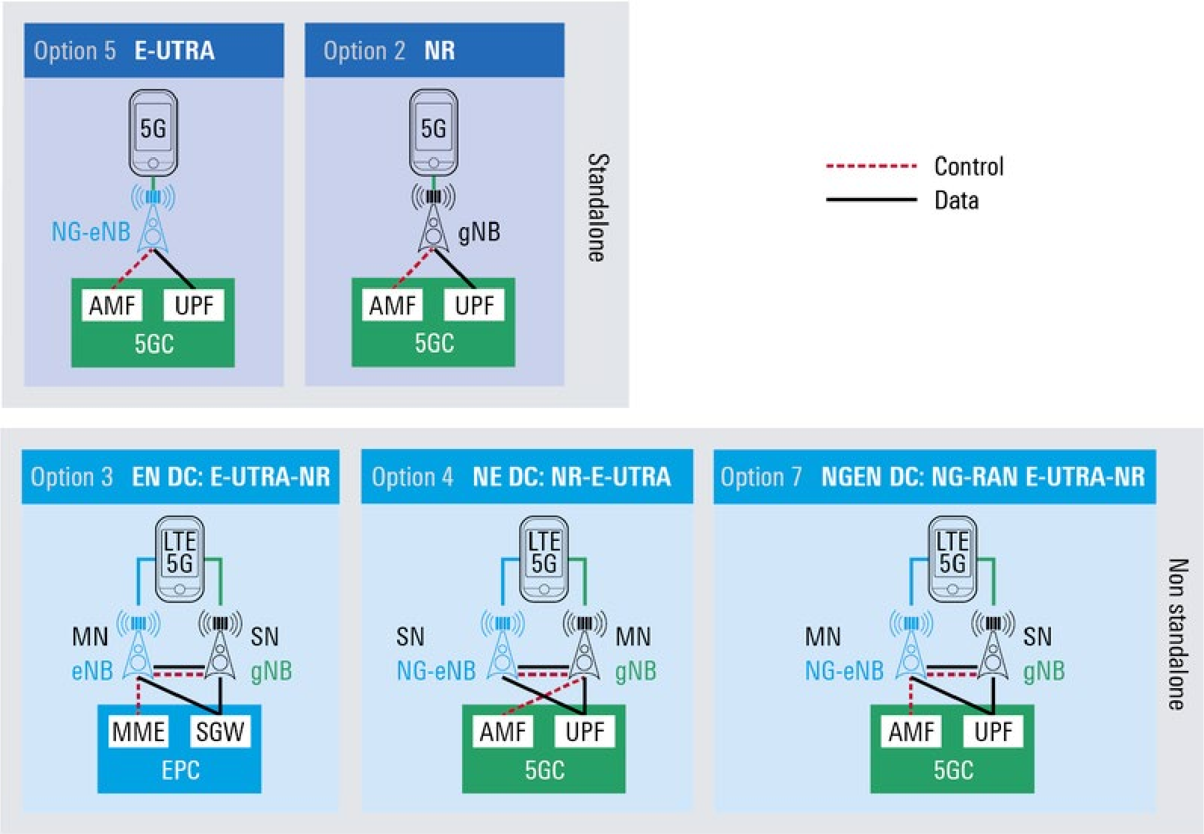


Figure 2-7 Connectivity options with 5G-NSA and 5G-SA[16]

The user plane is where data is both uploaded and downloaded, up- and download latency is asymmetric and the upload latency is estimated to 17ms and the download latency is estimated to 12.5ms according to [17]. These latencies are in addition to the latencies that naturally occurs on the internet to reach an endpoint (e.g., a website).

When a vehicle is travelling, the user equipment (UE) will need to switch between cells to get the best possible performance. LTE is set up in a way that cell change will use a "break-before-make" pattern. This will result in a loss of connection for a short period of time before connection to a new cell is established. This results in a period where data is buffered which causes a handover latency. In [18], physical tests in an area in city center of Aalborg and on a highway near Aalborg have been performed, where the average handover time is in the range of 24.5ms to 28.9ms, with extrema points up to 100ms.

The most important radio signal measurements within cellular [19]:

- Reference Signal Received Power (RSRP) or in common words signal strength as it is measured in dBm
- Reference Signal Strength Indicator (RSSI) is total power received from all cells over the entire bandwidth. Measured in dBm
- Reference Signal Received Quality (RSRQ) are calculated based on the formula in (2.1), where N is the number of resource blocks used. Measured in dB
- Signal to Interference or Noise Ratio (SINR), used as an indicator of signal quality and coverage. Measured in dB

$$RSRQ = \frac{RSRP}{RSSI}N \tag{2.1}$$

3 Use cases and services

In chapter 2, an overview of the standardization for C-ITS within ETSI is given and an overview of the Nordic Way interchange architecture which is one possible basis for being able to realize C-ITS communication over cellular network. Standardization is an enabler for communication between vehicles, and thus also for the use case of warning about wrong way drivers.

This chapter is to present different wrong way ahead systems. In chapter 3.1, the system installed in Knappe tunnel for detecting wrong way drivers is described and a real-life episode with a wrong way driver is presented. Chapter 3.2 describes the use case defined by the C-Roads platform. Chapter 3.3 withdraws the key elements from two different standards published by ETSI. Chapter 3.4 looks at other related work relevant for wrong way driver use cases.

3.1 Knappe tunnel

The Knappe tunnel is a 5.4 kilometer long, four-lane, twin tube, subsea tunnel on county road 557 in Bergen municipality. It was built in two stages, where second stage was finalized in 2015. It has two extra entrance and exit segments, in addition to the entrance/exit in the ends, giving the tunnel some extra traffic complexity.

The Automatic Incident Detection (AID) system in Knappe tunnel is a camera-based system that detects traffic incidents as stopped vehicles, vulnerable road users (e.g., persons, horses etc.) and wrong way ahead vehicles. Such a system is required by law on tunnels above a certain length and traffic[20, 21].

The cameras used for detection is placed with approximately 60-65m between each. Each AID-camera can detect a wrong way ahead vehicle and will subsequentially raise an alarm in the Traffic Management Center (TMC).

When detected on two subsequent cameras, the AID confirms the wrong way ahead incident. This to avoid false detection of incidents and thus avoid reduction of the trust in the system. When a wrong way ahead is confirmed by two cameras the following happens automatically:

- Tunnel closes to avoid more vehicles entering a potentially dangerous situation
- Speed limit is reduced from 80 km/h to 50 km/h
- Left most lane is closed by lane control signs and all vehicles is directed over to the right most lane
- Emergency evacuation lights is ignited

In short words, the system is an automatic infrastructure-based Wrong Way Ahead system that warns vehicles by reducing the speed limit, closing the left most lane and closing the tunnel.

The latency in the system is based on how fast the two subsequent AID-cameras detect a vehicle heading the wrong way, i.e., how fast a vehicle passes the two AID-cameras.

3.1.1 Real life wrong way driver episode

In this chapter, an episode involving a wrong way driver will be presented to provide insight about how a real-life scenario will take place. The episode is from 2016 and is recorded in Knappe tunnel. In the upper left corner of the images in Figure 3-1 to Figure 3-7, the white text is the camera number, and the yellow number is the seconds from the video started.

In Figure 3-1, we see a car entering the tunnel on one of the entrance ramps, braking and using its blinkers to signalize that it is going to turn left on to the traffic direction. In Figure 3-2 the car is passing the road markings and in Figure 3-3 it is driving across the right lane, blocking the road for an approaching car that barely avoids a collision. In Figure 3-4 it is over in the left lane.

In Figure 3-5, the car has entered the zone for the next camera and this camera triggers a "Wrong Way Driver" warning, indicated by the text "INV DIR 2" in Figure 3-6. The automatic warning of other vehicles is triggered in Figure 3-7, where the speed sign and emergency evacuation lights has been switched on.

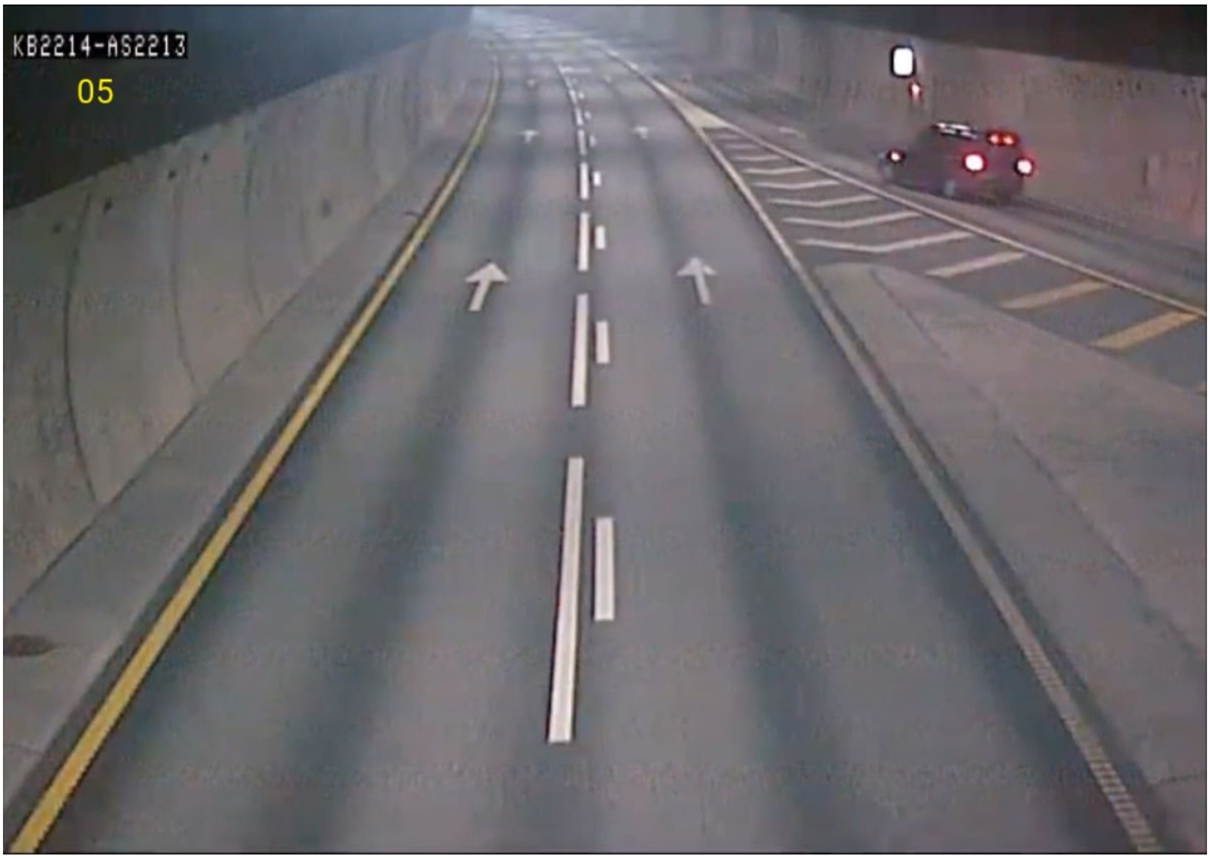


Figure 3-1 Car slowing down on entrance ramp and uses blinkers to signalize it is going to turn left

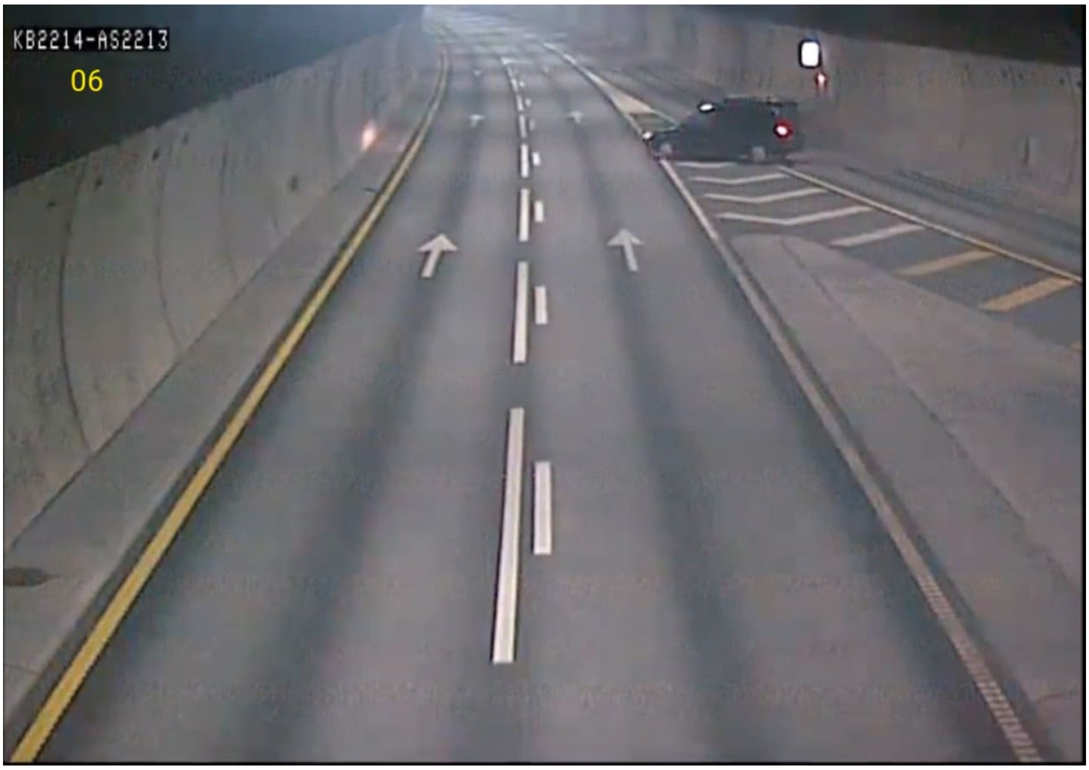


Figure 3-2 Car about to pass the road marking

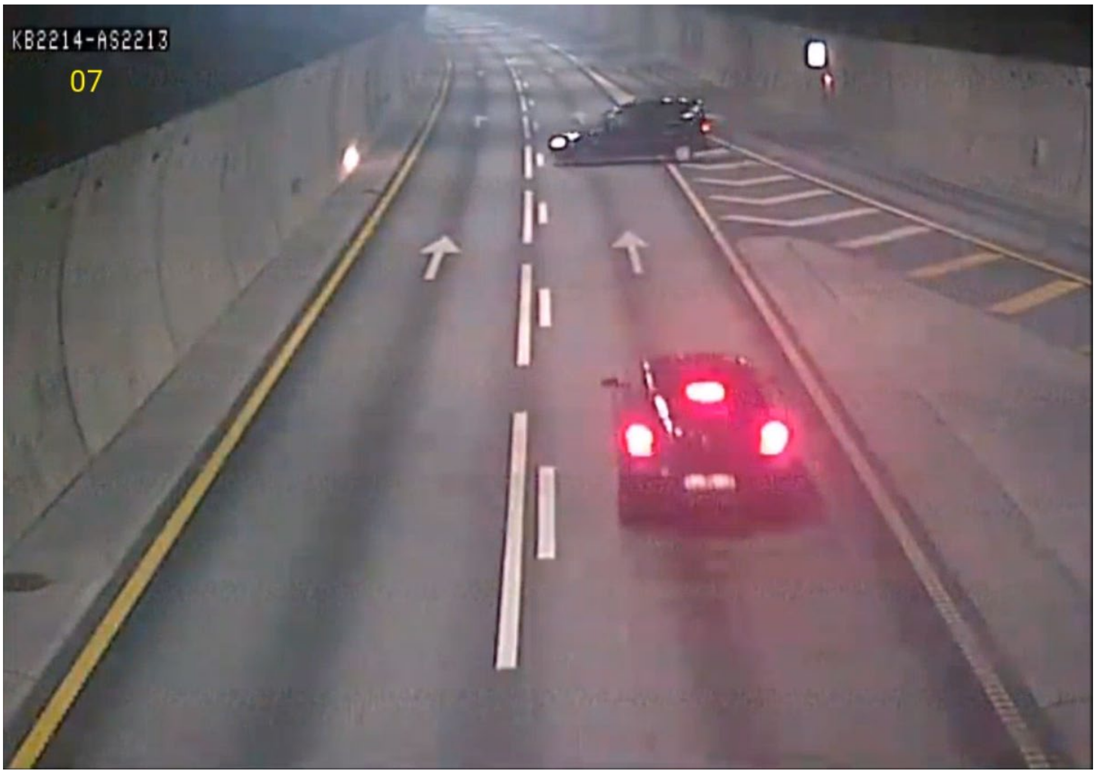


Figure 3-3 Near collision avoided by the emergency breaking of the approaching car

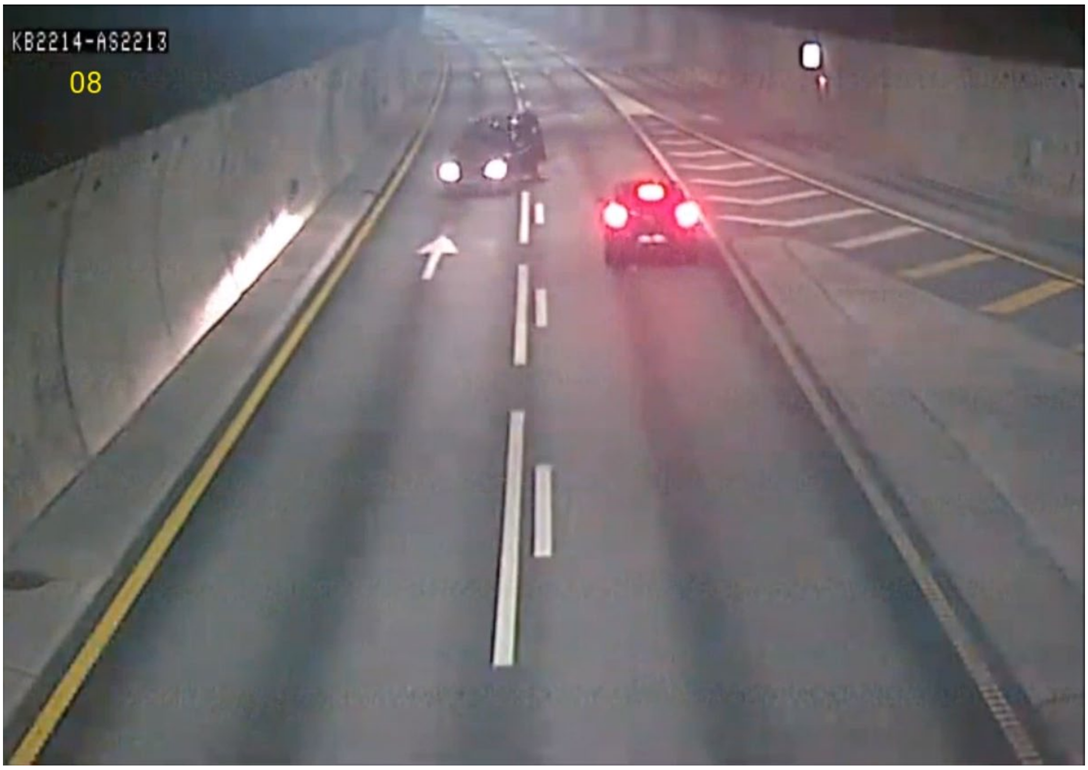


Figure 3-4 Collision avoided, and the wrong way car is over in the left most lane.

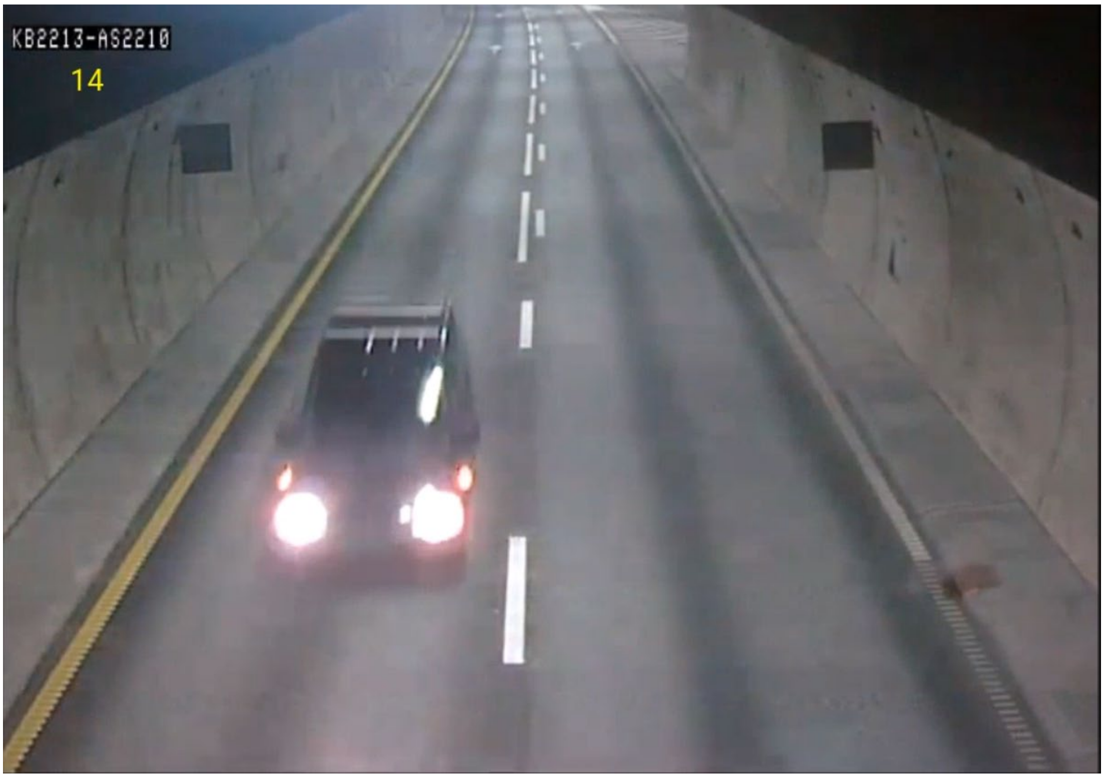


Figure 3-5 Car have entered the zone for the next camera and is using emergency flashers, probably to signalize to others that it is going wrong way.

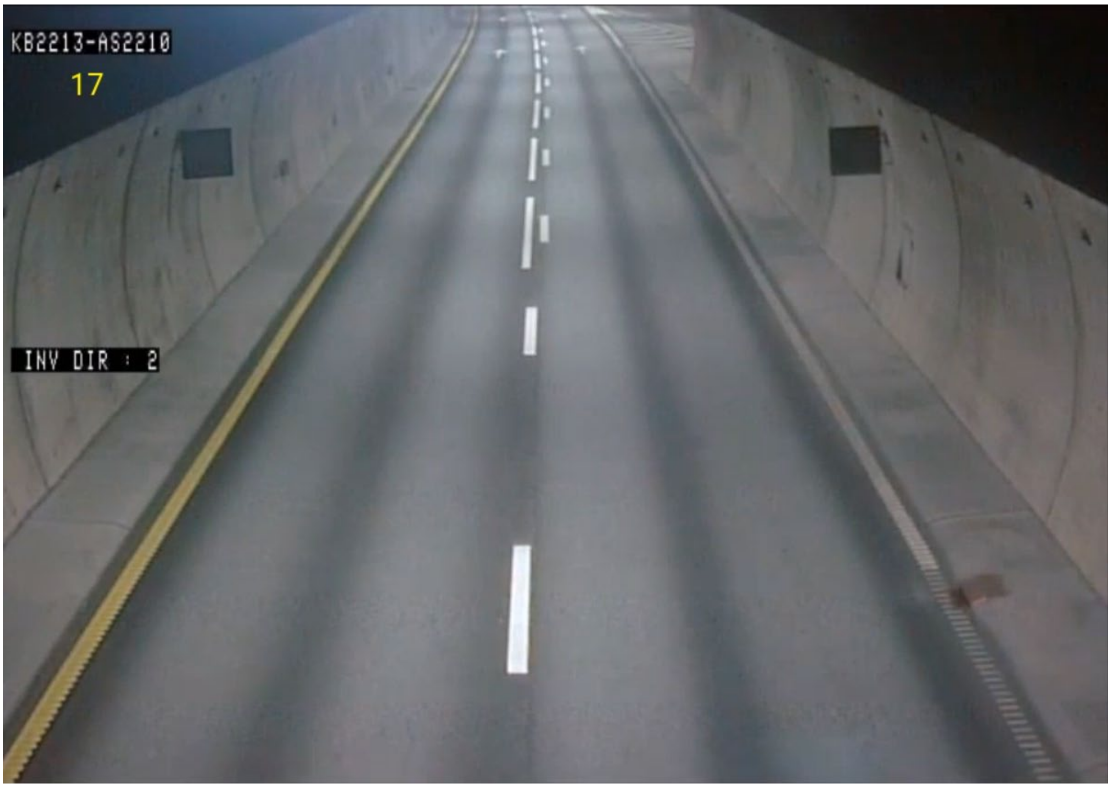


Figure 3-6 Inverse direction is triggered by AID-system

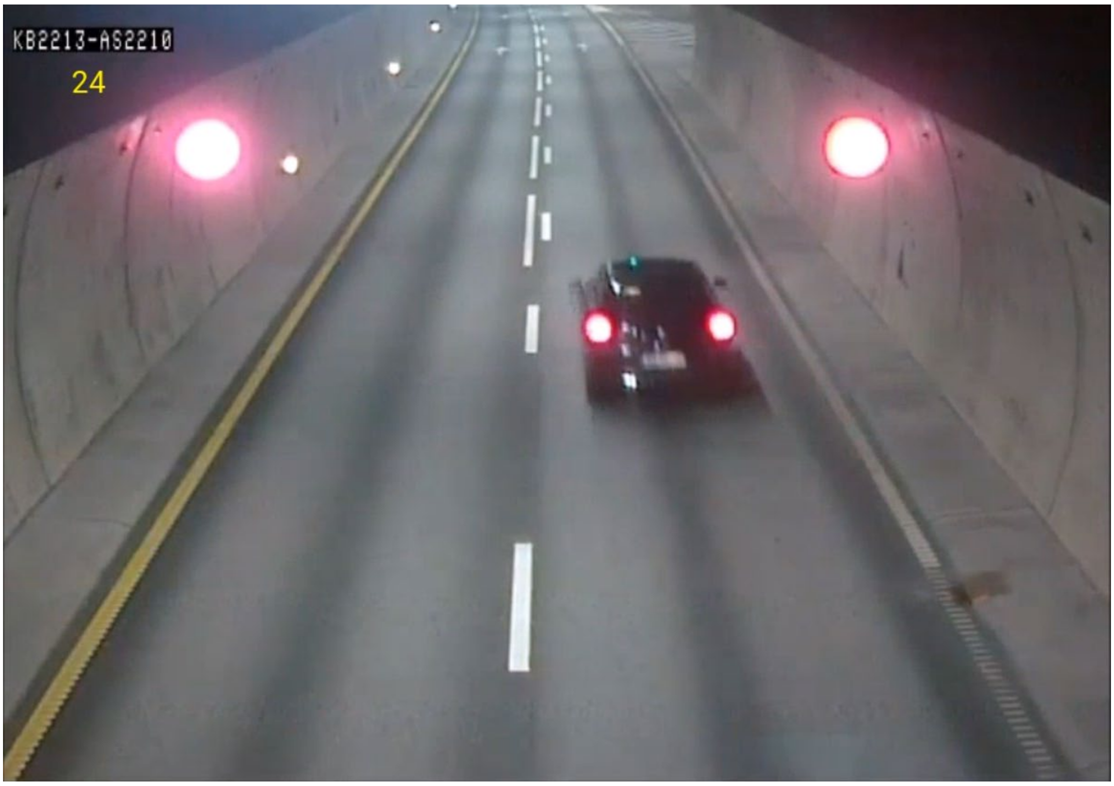


Figure 3-7 Warning of other drivers by lowering the speed limit and igniting emergency evacuation lights

From this episode, it takes about three second from the car starts to turn on traffic to it is over in the left field, and it takes 19 seconds before the automatic warning of other vehicles starts.

This includes the latency in the AID system and tunnel control system that controls the signs and emergency lights.

3.2 C-roads - Hazardous Location Notification - Alert Wrong Way Driving

First of all, it is good to know what the C-roads platform is[22]:

> *The C-Roads Platform is a joint initiative of European Member States and road operators for testing and implementing C-ITS services in light of cross-border harmonization and interoperability.*

The C-Roads platform have specified a service and use case called Alert Wrong Way Driving (AWWD) and sorts under the umbrella of Hazardous Location Notification's (HLN)[22].

It describes several possible detection methods, like camera-based AID, automatic wrong way detector (e.g., inductive loops in the road), manual reporting (e.g., by phone call from personnel having a visual detection) and "Other C-ITS equipped vehicles". What is meant by the "Other C-ITS equipped vehicles" clause is not elaborated, but it is possible that C-ITS enabled vehicles can report about AWWD based on the vehicle's sensors, e.g., camera, radar, lidar etc.

The scenario for this use-case, describes a "two-stage-human-in-the-loop", where an operator at a TMC first is notified by an ongoing situation, and performs an action by sending out a DENM to vehicles. In the second stage, the TMC-operator gets confirmation on the ongoing situation from a second source and based on this information updates the DENM with a higher priority than earlier.

The messages specified in this use case is to transmit a DENM with causeCode: 14 (wrongWayDriving) and subCauseCode: 2 (wrongDirection) within the Situation container as specified in chapter 2.2.3.

If this use case is used in IP-based communication, like the Nordic way interchange in chapter 2.3. It is pre-defined to use serviceType HLN-AWWD and messageType DENM in the application properties to allow for filtering of messages as mentioned in chapter 2.3.

There are no latency requirements in the service or use case definition, but with a "human-in-the-loop", substantial latency is added.

3.3 ETSI Wrong way driving use cases

Within the ETSI standards there is identified two standards that describe the use case of wrong way driving.

In [23], the use cases is focused to be used over short-range communication (ITS-G5) and describes many services within traffic safety, traffic efficiency and other services. Under traffic hazard warnings, wrong way driving is listed. It gives a short description about the aim to warn other drivers, and list four main requirements. The requirements are that the vehicle driving wrong way must be able to detect and broadcast to other vehicles with DENM's that it is driving wrong way, similarly the concerned vehicles need to be able to receive and process these DENM's. The requirements state a minimum frequency of 10 Hz message dissemination with max latency of 100ms.

[24],[25] and [26] is a three part standard that describes functional requirements for Road Hazard Signaling (RHS). The first part of the standard lists 10 different RHS and specifies general functional requirements, common for all RHS use case, and requirements for each individual use case, among them "Wrong way driving". It seems to build on the use case in [23] as the requirement seems to be rather similar, but are more detailed with CauseCodes and SubCauseCodes for the data elements in the DENM's.

Notable requirements are the maximum latency requirement of 300 ms in Figure 3-1. It shows 7 timestamps (T0 – T6), the standard defines a maximum latency between T1 and T5 to be 300ms. It also differentiates between two performance classes, Class A and B, where it in Class A shall be less then 150ms latency between T0 and T1. Whereas for Class B shall be less than 1,4s.

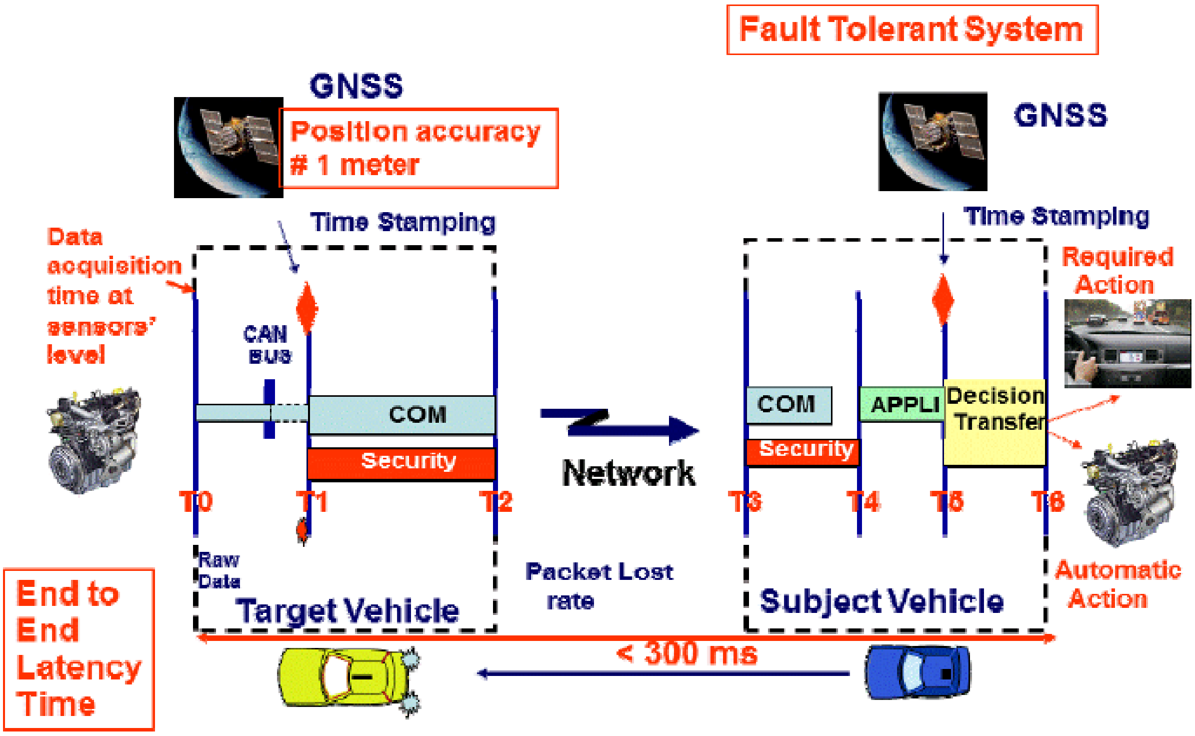


Figure 3-8 End to end latency requirments for collision avoidance[24]

Another notable requirement is the requirement of sending DENM-messages at up 10 Hz rate.

For an ITS-station to notice if a vehicle is driving in a countersense direction, it needs to know the position and trajectory of the vehicle, which it will get from a CAM-message. It then needs to match this position and trajectory to a map that holds the information about the road network with actual drive directions, commonly denoted as Local Dynamic Map (LDM). This way, it can detect a wrong way driver based on information from the CAM-message and map matching.

3.4 Other related work

Aventi did a test on wrong way driving sending CAM-messages from a vehicle to an Road Side Unit (RSU)[27]. This test used short range communication (ITS-G5) to transfer the message from the vehicle to the RSU, where the logic for handling the “Wrong way driver” is programmed. The aim for the test was to verify the possibility to get an incident from the road published on NPRAs DATEX node. Although everyone can request information from DATEX, it is not standardized under the ETSI-standards.

In [28], the authors have studied the communication performance requirements for various levels of C-ITS and Connected, Cooperative and Automated Mobility (CCAM). It has been categorized into four categories: end-to-end latency, reliability, data rate per vehicle and communication range. The figures for cooperative awareness state a maximum end-to-end latency of 100ms with 90-95% message reception reliability and a data rate between 5-96 kb/s. The communication range is mostly important when comparing short-range communication methods.

[29] have identified the most critical performance indicators for CCAM and have included cooperative awareness as one of the use cases under CCAM. Three of the indicators is the same as the one in [28]; latency, data rate and reliability. The fourth requirement is a bottleneck requirement for stress on the communication network, e.g., if there is a traffic jam with multiple vehicle communication through the network. The indicators for cooperative awareness, sensing and maneuvering are: data rate of 5-25 000 kbps, latency within 5-1000ms and a reliability of 90-95%.

[30] tests the latency over LTE Advanced supported by edge computing, considering three use cases, where the common factor is the need for low latency. Transfer of data is tested using CoAP, which uses UDP/IP, and MQTT, which uses TCP/IP. The test is performed by sending 56 bytes with 20 ms interval, to the CoAP and MQTT-brokers. In the LTE Advanced test, the brokers were placed in the LTE Core network (EPC). The results show that CoAP performs better then MQTT, but the results are a bit peculiar as MQTT has about ten times the latency of CoAP, and that MQTT reduces the latency with 50 % when it is using 10 publishers instead of 1.

4 Test and test system

This chapter describes the test system used collecting metrics from field trials. The system is based upon the test system developed in [31], which was done in a pre-project for this project. A good basis was made for making a test system for several services, but it has been adopted to serve for the specific need in this project. In addition, some enhancements have been done to get better code and functionality.

4.1 Cloud system

In Figure 4-1, the overall system is displayed. The aim of this system is that it replicates the key functionalities in the architecture defined in Figure 2-1 and an important part of this architecture is that the backend service runs in a cloud environment.

The cloud system runs as a container based environment in Kubernetes in Google Cloud on zone “europe-north1-c” which is situated in Hamina, Finland[32]. The system running in Kubernetes have been configured using the code in [33] and the following is running:

- An elastic cluster[34] is used for storing logs from the containers in the Kubernetes cluster and logs from the onboard unit which is uploaded manually.
- Nginx ingress controller[35] for controlling the routing from the endpoint to the correct container. For pure TCP-traffic, as for the MQTT-protocol, nginx routes directly on port 8883 to the mosquitto MQTT-broker.
- Mosquitto[36] is the MQTT-broker used in this system, it is configured to use secure communication of TLS and authentication of the onboard unit is done with certificates.
- The backend service is written in node.js and is further explained in chapter 4.3

The two other components shown in Figure 4-1, are the Nordic Way Interchange and the onboard unit situated in the vehicle. These are described in chapter 2.3 and chapter 4.2 respectively.

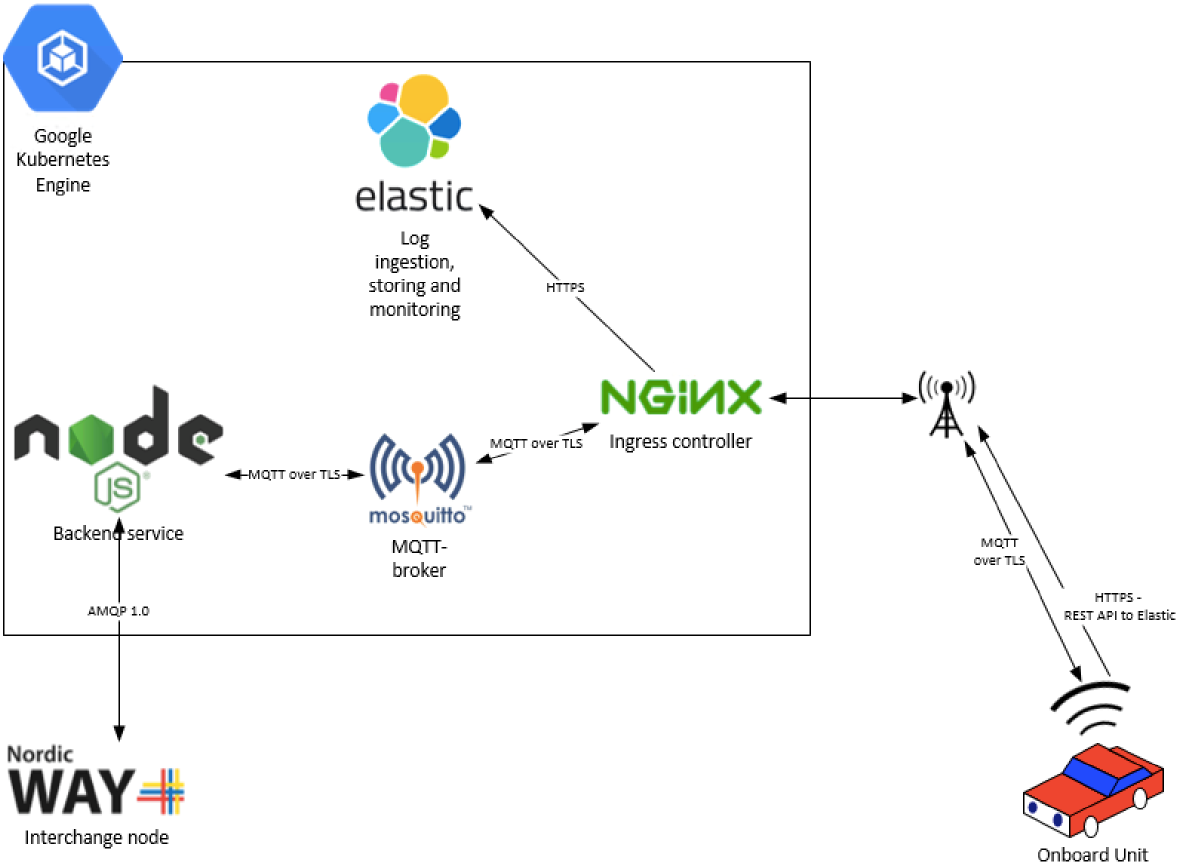


Figure 4-1 Overall test system showing the connection and protocols between the different components in the test system.

4.2 Onboard unit

In the pre-project [31], the OBU was programmed in a python program that collected data from a GNSS receiver and an external cellular router. The sequence diagram in Figure 4-2 shows the message flow in the program that have been developed in the pre-project and adapted in this report. The code for this project is available in [37]. An important note is that the program in the sequence diagram is asynchronous, except of the logging module. This way, there is no blocking of the event loop if it occurs waiting time, which is common in network operations.

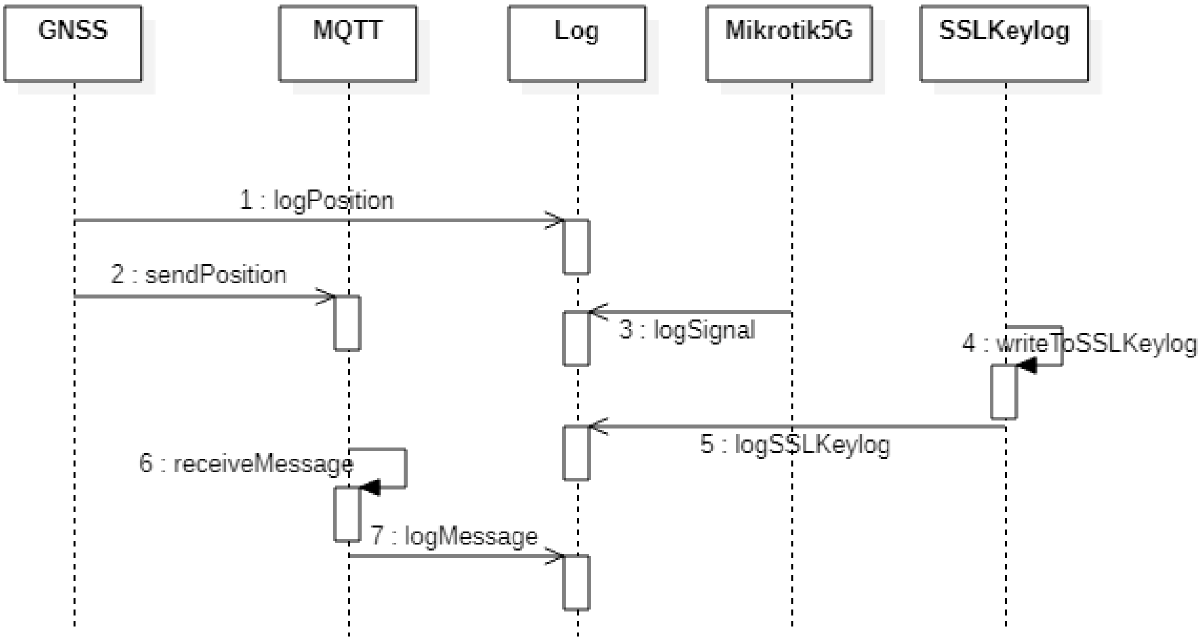


Figure 4-2 Onboard unit sequence diagram

Like in the pre-project, the python program runs in a docker container, abstracting the code away from the OS. This gives a system abstraction like the one in Figure 4-3. This way it is easy to isolate the code and docker can also handle restarts it the program crashes.

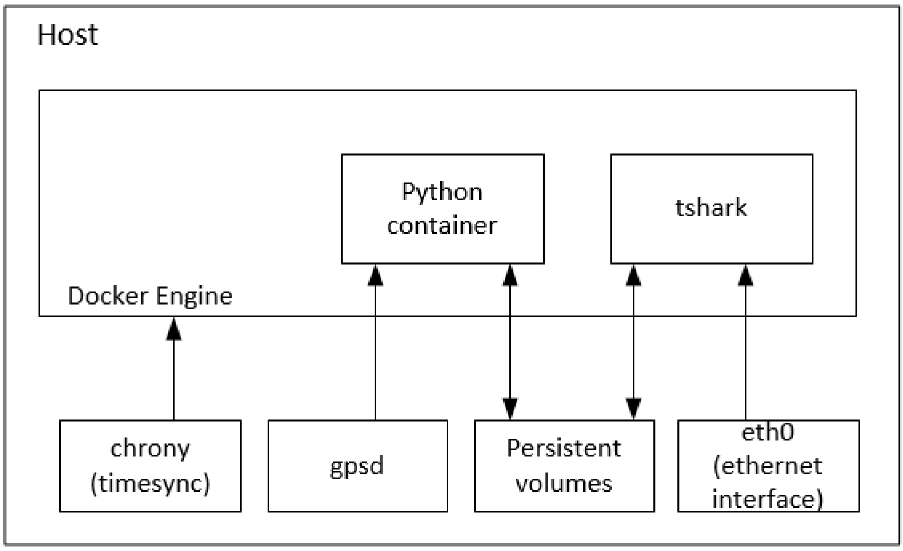


Figure 4-3 Onboard unit - system abstraction

The communication protocol is still based on MQTT as it is a lightweight protocol that is easy to implement. MQTT has the possibility to send messages with different Quality of Service (QoS). For QoS 0, a fire-and-forget pattern is used, meaning that the MQTT will only transmit the message once. For QoS 1, the message will be delivered at-least-once and for QoS 2 the message will be delivered exactly-once. For a moving vehicle, its position is “outdated” as soon as a message is sent, so QoS 0 is selected used for this project. QoS 0 will thus potentially result in some packet loss.

This program is mostly reused from the pre-project, but some changes and improvements have been done to improve the code:

- The collection of metrics from the cellular network router is now using aiohttp[38] instead of requests[39] to get use of asynchronous functions. Even though the latency of retrieval from a local network device is rather small, it is good practice to use async when retrieving information over a network connection. Likewise, positional data from gpsd[40] is now using gpsd's asyncio python library to read data from gpsd as it communicates over the internal loopback interface.
- To be able to decrypt the packets captured by tshark mentioned in chapter 4.1.4, it is necessary to supply tshark with a key from each TLS session that is established. This is retrieved with sslkeylog-library[41] that writes the sslkeylog-line to a file every time it changes.
- Shutdown and proper error handling have previously been neglected, but is now overhauled based on Lynn Root's guide[42]. Now the code shuts down on signals or if an error occurs, making sure that the shutdown becomes graceful and that all connections are closed. The shutdown on error is intended, since docker compose will restart the container.

The logs collected from this program is stored locally, and later being preprocessed and uploaded to the elastic cluster in Kubernetes for log storage using a separate program written in python[37]. It would have been possible to stream the logs directly to the elastic cluster during a field trial, but it might impact the metrics gathered as it would consume on the same cellular network connection.

4.2.1 Physical setup of onboard unit

Figure 4-4 shows the physical connections for the OBU. The OBU is based on a Raspberry Pi 3 Model B running the Raspbian OS and is connected to a 5G router and two GNSS receivers. The Raspberry Pi and 5G router are connected to the cars power grid, while the GNSS receivers are powered from the Raspberry Pi.

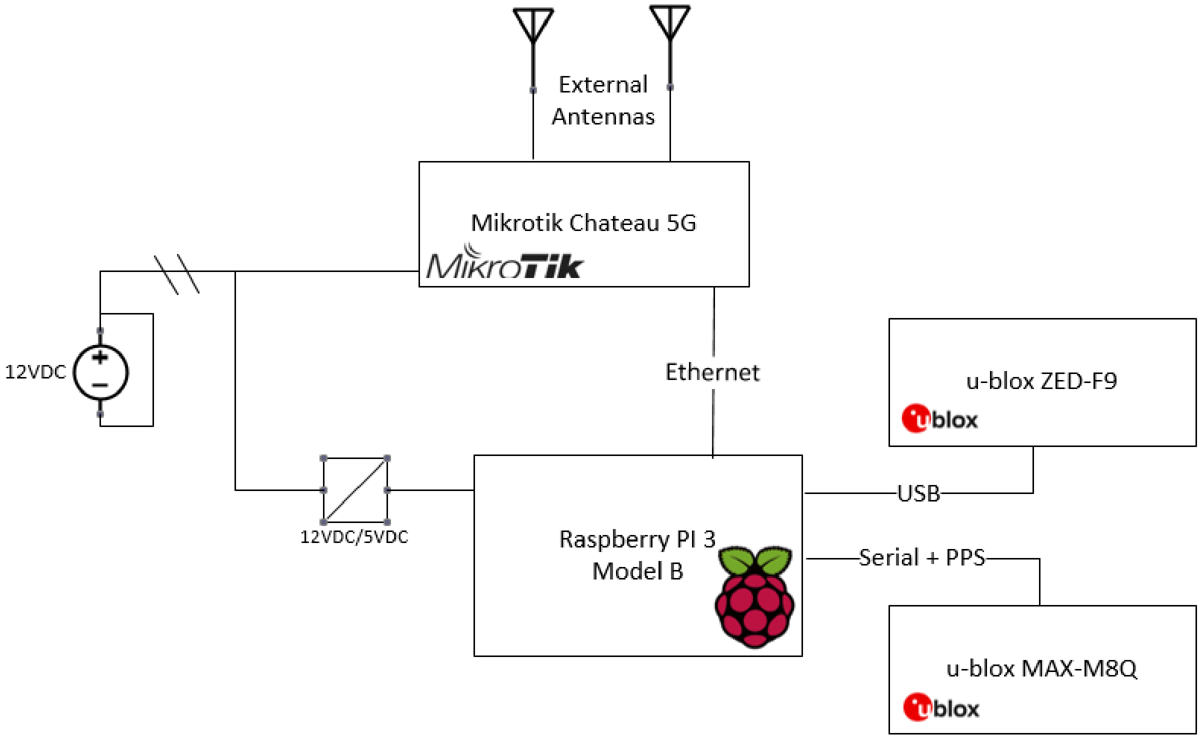


Figure 4-4 Onboard unit - physical setup showing the connection between Raspberry Pi, two u-blox GNSS-receivers and the Mikrotik 5G router.

4.2.2 Mikrotik 5G router

In the pre-project[31], the router was a 4G router and have now been exchanged to a 5G router. The new router is a Mikrotik Chateau 5G [43] with external 5G antennas and can be powered on a voltage from 12-28 V, making it suitable for connecting to the power grid in a vehicle. It is a consumer grade 5G router, but it is possible to retrieve information from the router by using a built in REST API. With the API, it is possible to retrieve the same information as from the web interface or SSH session and log information as signal strength, frequency band etc.

It is equipped with external antennas, that with a cable extender can be placed on the roof of a vehicle. Measurements from within a vehicle will be affected negatively, since a vehicle is like a faradays cage that blocks signals effectively. Therefore, it is desirable to place the antennas outside the vehicle.

The router is equipped with an ordinary SIM-card from Phonero, which uses Telia’s cellular network.

4.2.3 Time synchronization with GNSS as source

From chapter 2.4, we have that it is different upload and download latencies in a cellular network. To measure both the upload and download speed, the reference times needs to be synchronized to the same reference.

The use of public available NTP servers over network connection can typically give an accuracy of ±30mSec and can be influenced to the worse by the network connection[44]. In the case for the test scope of this project, the network connection can potentially be flaky during trials and might impose unnecessary inaccuracy on the time synchronization.

A good and cheap solution to get a high precision time synchronization is to use a GNSS receiver with a Pulse per second signal (PPS). In this system a Waveshare MAX-M8Q GNSS-extension board[45] that stacks on top of the 40pin GPIO on the Raspberry Pi is used, see Figure 4-5. This extension board is based on the u-blox MAX-M8Q-receiver[46] and have a PPS-output. It is equipped with a dedicated battery for persisting the configuration in the RAM. For best performance with the receiver, the dynamic model is set to automotive[47] as it will travel along the road. This mode assumes low vertical acceleration and shall in theory give better GNSS-positioning.

The PPS gives a pulse exactly when a new second starts. Given that it only gives a pulse, it is not possible to know the time from the PPS alone, but together with the time reported from GNSS-receiver, it is possible to get a high precision time synchronization on the Raspberry Pi. The PPS-output is attached to GPIO-pin 18 on the Raspberry Pi. This pin is designated for working with clock signals and works in an interrupt like manner. The signal interacts with the kernel-module pps-tools[48]. To merge the timestamp from PPS and GNSS, chrony[49] is used. chrony is both an NTP-server and client, and can use other reference clocks, as used in this case where GNSS is used. It then manages the system clock and synchronizes the system clock to the best available source. The installation of chrony with PPS have been done on basis of the guide in [50].

The time zone is chosen to be UTC 0, as this harmonizes the time zone across the entire test system and helps to avoid misunderstandings later.

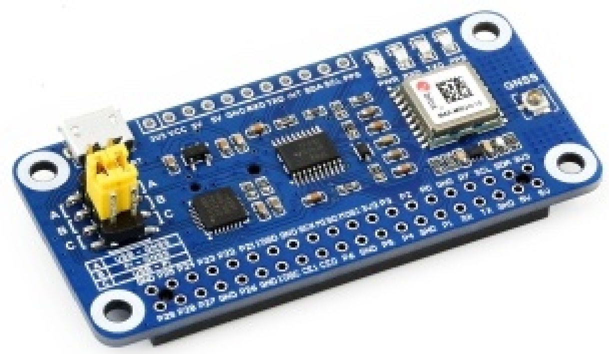


Figure 4-5 Waveshare MAX-M8Q GNSS-extension board used for time synchronization over GNSS[45]

As pointed out in [31], Google handles leap seconds with leap smearing[51]. While this would impose a potential issue in a system running 24/7/365, it is not an issue here as the test will be performed over a relative short period. It will not be added an extra leap second this year and next chance for adding a leap second is 30.06.2022[52], so leap second won’t be an issue.

4.2.4 Vehicle positioning

For positioning, a SparkFun ZED-F9R GPS pHAT is used together with an active antenna[53]. This board uses the u-blox ZED-F9R receiver and is a professional grade GNSS receiver. It is equipped with an IMU and input for wheel ticks and direction to use with dead reckoning. Since it won’t be possible to connect the wheel tick and direction to a production car that is to be used in a trial, this functionality won’t be used in this project.

According to the datasheet[54], the receiver has an accuracy of 1,5 m Circular Error Probable (CEP) over 24 hours measurement with a static antenna using satellites only. The datasheet doesn't state the confidence interval for the CEP, but is assumed to be 50 % based on [55].

The accuracy can be further improved by using Real Time Kinematics (RTK), but this requires a fixed base station that can send correctional messages to the ZED-F9R. For vehicles, this means that it needs base stations placed within a radius of 10km[56] for every location of the vehicle. The GNSS-device in the vehicle can then receive correction messages from the base stations to achieve very precise localization. There exist standardized message formats for these correction messages, where RTCM is the common one.

Transmittal of RTCM messages can be done by several protocols, where NTRIP is a common one when transmitting over internet. This protocol can be used for vehicles, but there also exist protocols that are specifically made for ITS-purposes.

For ITS-purposes, "GNSS Positioning Correction (GPS) service" has been standardized in [9, 57]. This service sends out RTCM messages and uses the same ITS-architecture as defined in chapter 2.1.

The cellular network can also transport RTCM messages in addition to over ordinary IP-based networks. In 3GPP Release 15, the use of GNSS corrections was included in LTE Positioning Protocol and can use the cellular network in a standardized way to send correction messages to UE[58].

Hence, it would be possible to improve the positioning, but the solution without RTK and dead reckoning is adequate for this project.

4.2.5 tshark

To log all packages on the ethernet interface of the OBU, tshark[59] is used. This is a command line version of the more known Wireshark[60] program that operates with a GUI. tshark sniffs the messages that are on the ethernet interface of the Raspberry Pi, and thus doesn't interfere or delay the traffic in the system.

The use of tshark serves several purposes:

- It tells when the packet is transmitted and received on the network interface, giving a chance to calculate the delay in the python program.
- It tells what other traffic runs over the interface, adding the possibility to verify that the main part of the traffic is used for the transmitting messages to the backend service
- MQTT relies on TCP/IP and TCP packets can be retransmitted in case it doesn't get an acknowledgement of the TCP-package within a variable time interval. The packets logged by tshark can be opened in Wireshark to analyze for retransmits

tshark runs in a container that starts together with the python container and logs the packets to a folder on the host as visualized in Figure 4-3. Code for tshark in a container is forked[61] and adapted to the needs for this system[62].

Since the MQTT-packets is encrypted when captured by tshark, there is a need to post-process the logs to decrypt the packets before uploading them to the log storage using a separate python program[63].

4.3 Backend service

The backend service has been completely rebuilt compared to the one used in the pre-project [31], using a backend server based on node.js instead of python. This due to node.js' asynchronous nature, opposed to the lack of an AMQP 1.0 library compatible with pythons asyncio standard library.

The message flow is rather simple and visualized in Figure 4-6. Basically, it just receives the "CAM"-message from the MQTT-broker, logs it and sends a "DENM"-message to the NWIXN. It then receives the same DENM-message from NWIXN, logs it and sends it to the MQTT-broker. This way the messages is logged with a timestamp to the elastic cluster and can be retrieved later.

Following libraries are used in addition to the standard node.js libraries

- rhea AMQP 1.0 library[64] for connecting to the NWIXN
- mqtt-library [65] for connecting to the MQTT broker
- pino-library [66] is used for logging in json format

The code for the backend service is available in [67]

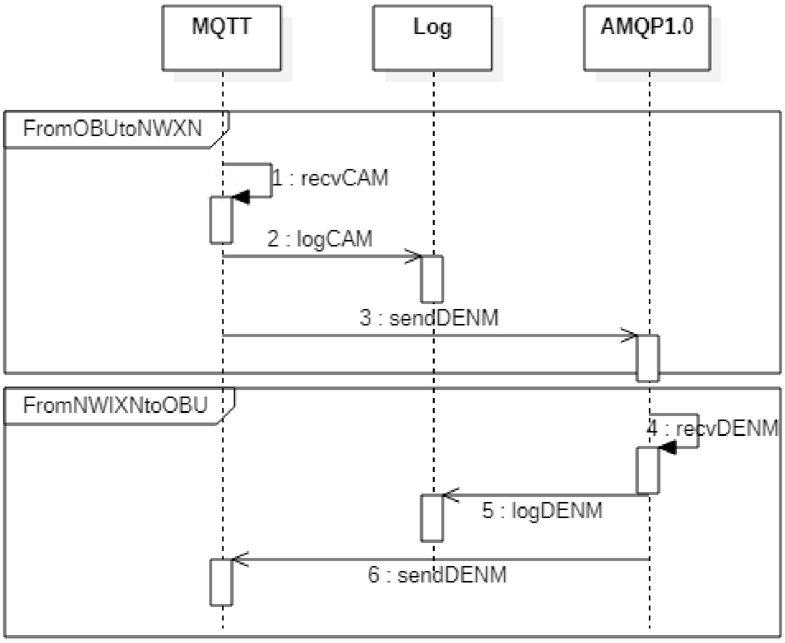


Figure 4-6 Sequence diagram for backend service

The time synchronization is assumed to be precise, as the Kubernetes cluster is synchronized within the Google Cloud datacenter.

Ideally, this backend service should be compliant with the ITS-station architecture described in chapter 2.1 since it shall send ASN.1 UPER encoded messages which is signed, and thus decode and verify signatures. It should also have implemented the Decentralized Environmental Notification service to generate DENMs. A part of this service is to hold a Local Dynamic Map (LDM), that can be used to match vehicles positions on the map. The LDM needs to hold the traffic direction of the roads to be able to map-match vehicles position and trajectory to detect if it is driving against traffic direction. All this will add processing delay compared to the solution that is implemented now, which is a minimal solution for this test system.

4.4 Field trial

4.4.1 Test parameters

The messages from the OBU to the backend system relies on MQTT for sending and receiving messages. Since the size of CAM and DENM can vary, a size of 400 bytes is chosen for CAM and 1000 bytes for DENM. This includes the size of security header described in chapter 2.2.1. In addition, the MQTT-header will add size to the total payload. Triggering of new messages will be done by the GNSS-receiver and it is limited to maximum trigger at 10 Hz but can be lower based on the signals received from the satellites.

4.4.2 Test area

The test is performed on E39 in Åsane in Bergen municipality. It is chosen due to being a divided highway with ramps, no tunnels and no tall buildings that might cause signal problems. It is also the area where the Wrong Way ahead-accident happened in [68] and [69]. The area for the test is on road E39 shown in Figure 4-7.

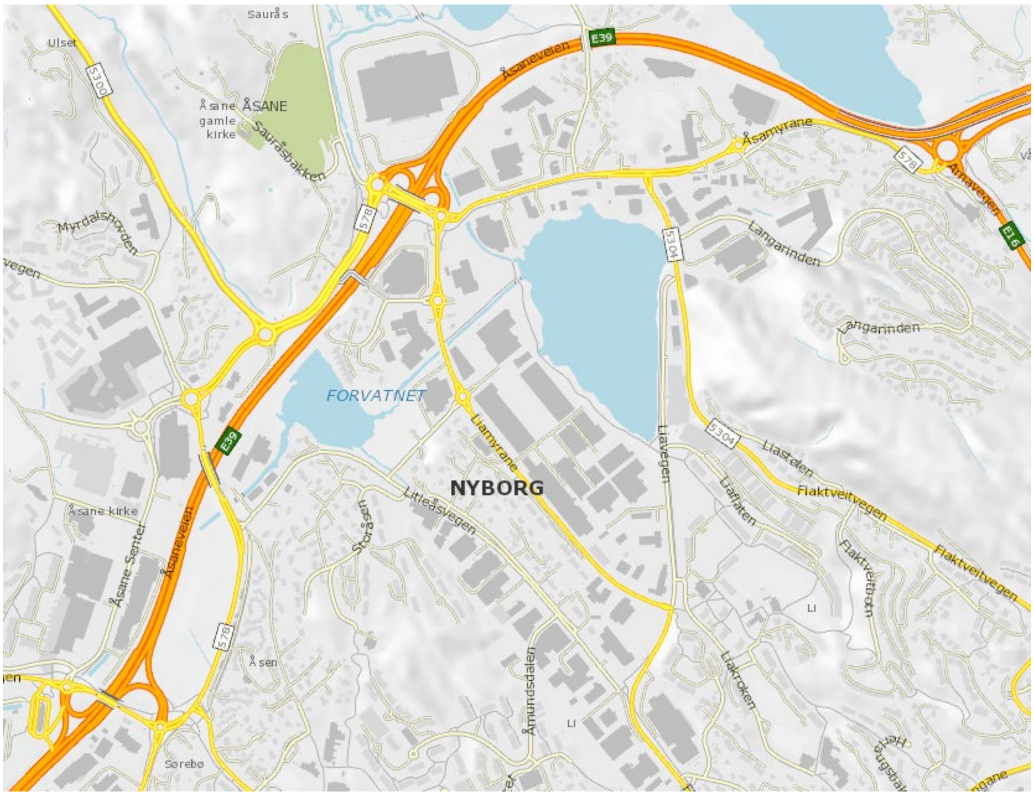


Figure 4-7 Test performed on E39 including ramps

According to Telia who is the cellular network operator for Phonero [70], the area has 5G coverage as shown in Figure 4-8.

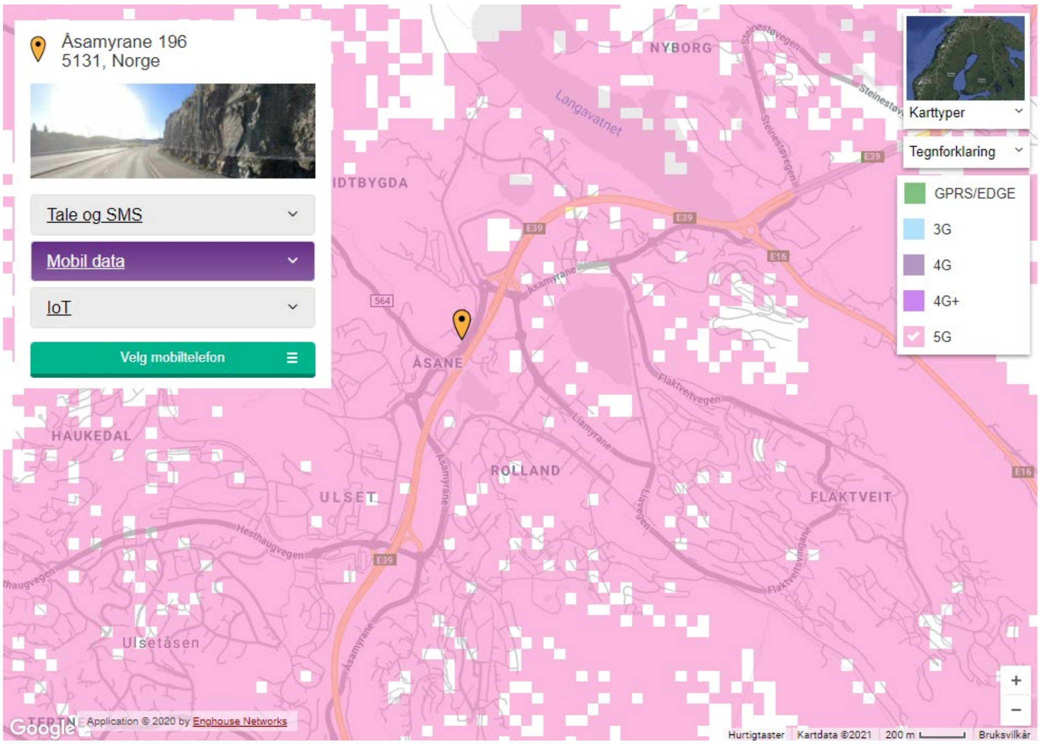


Figure 4-8 5G coverage map[70]

4.4.3 Performing the test

Performance of the test was done the 14.12.2021 and the route was driven three times to gather comparable datasets. The start and end time for each of the three tests are listed in Table 4-1

Table 4-1 Start and end time for the test. Time zone is UTC 0

	Start	End
A	09:30:29.419	09:39:46.662
B	10:04:28.509	10:14:13.819
C	10:22:10.660	10:31:47.200

The onboard unit from chapter 4.2 was rigged in a car, and antennas for both GNSS and cellular was placed on the car roof as shown in Figure 4-9. The figure shows three GNSS antennas in front, where only the two small one was used. In the rear, the two antennas for the 5G router are placed and secured to the bars with cable ties. The antennas are the original antennas that came with the router but is connected to a magnetic foot with a cable extender.

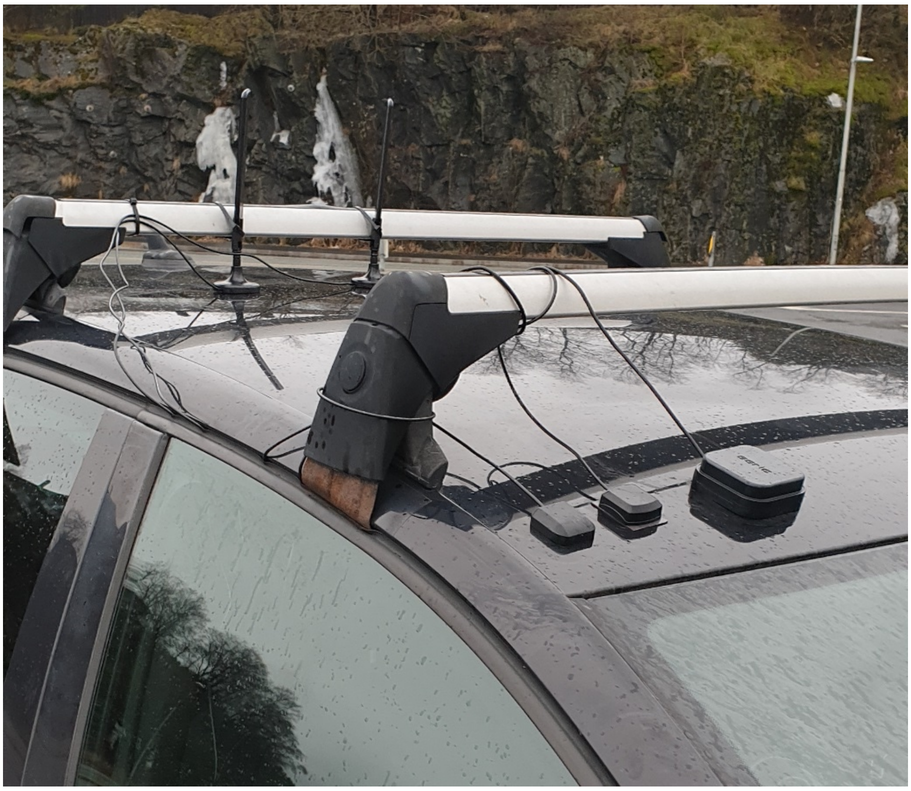


Figure 4-9 Placement of GNSS-antennas and cellular network antennas on the car roof

5 Analysis of data from field trial

5.1 Tools used for analyzing the data

These are the main tools used for analyzing of the data:

- VSCode [71] is an opensource integrated development environment (IDE) that supports plugins so it can be used with almost any programming language
- Jupyter Notebook [72] installed in a python virtual environment and with a plugin to VSCode
- Pandas [73] is used to structure the collected data and for performing the analyses
- Plotly.py [74] for plots and maps

5.2 Collected data

All data collected during field trial and a Jupyter notebook containing the code for producing this analysis is available in [75].

To easier understand where the different metrics is collected, a sketch is shown in Figure 5-1 to point out and name the measurement points. The naming of the green points will be used in figures and tables to easier explain what point the measurement originates from and have a concise naming of each measurement point.

- MpA to MpF logs the messages transmitted and received in the system. It logs the entire message which includes a unique id and a timestamp of when it was logged.
- Mt logs information from the Mikrotik 5G router. The important metrics logged and used in this analysis are:
  - Reference Signal Received Power (RSRP)
  - Reference Signal Received Quality (RSRQ)
  - Signal to Interference & Noise Ratio (SINR)
  - The number of Carrier Aggregation bands used by the router
  - Data-class (if it is 5G NSA or LTE)

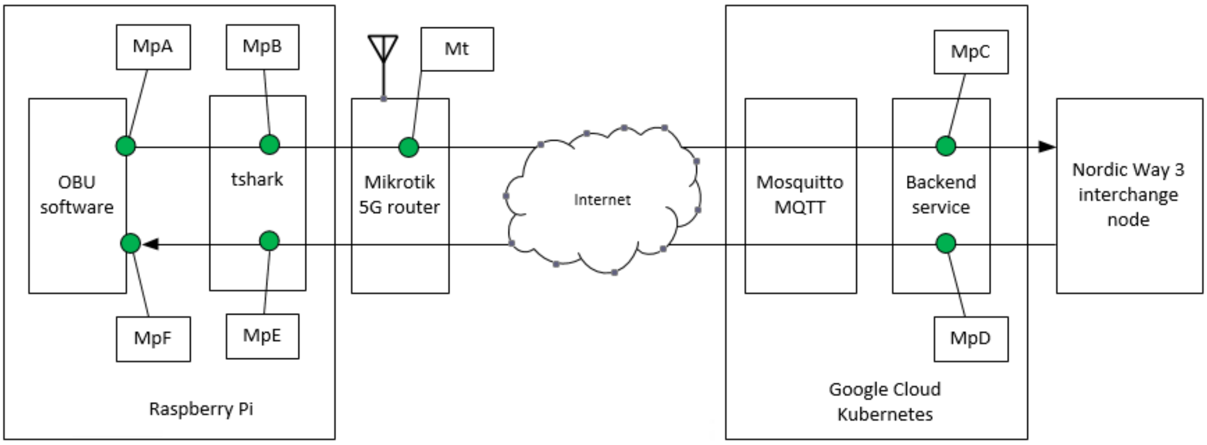


Figure 5-1 Measurement points in test system. Green point shows where the measurement is logged and naming of that point.

From the field trial, three sets of data were collected following the exact same route for a set of comparable datasets. Each data set are denoted A, B and C.

In Table 5-1, the number of collected metrics pr measurement point is listed.

- The logging of signals from the Mikrotik 5G router has been done at a 2 second interval, explaining why point Mt is substantial lower than the other points.
- Dataset C is missing data on MpB and MpE. This is due to that tshark was not able to decrypt the MQTT messages. This makes it impossible to concatenate with the rest of the data from dataset C. The dataset will still give value as the others measurement point will give insight.

Table 5-1 Number of collected data related to the measurement points in Figure 5-1.

	A	B	C
MpA	3142	3584	3599
MpB	3140	3574	
MpC	3142	3584	3599
MpD	3142	3584	3599
MpE	3142	3573	
MpF	3141	3579	3595
Mt	265	279	274

5.3 Latencies in the system

In Figure 5-2, the round-trip time for all messages is calculated for all three data sets. From the figure, it is easy to see some spikes of delay, especially for dataset B and C where the spikes are at 2,5-3 seconds.

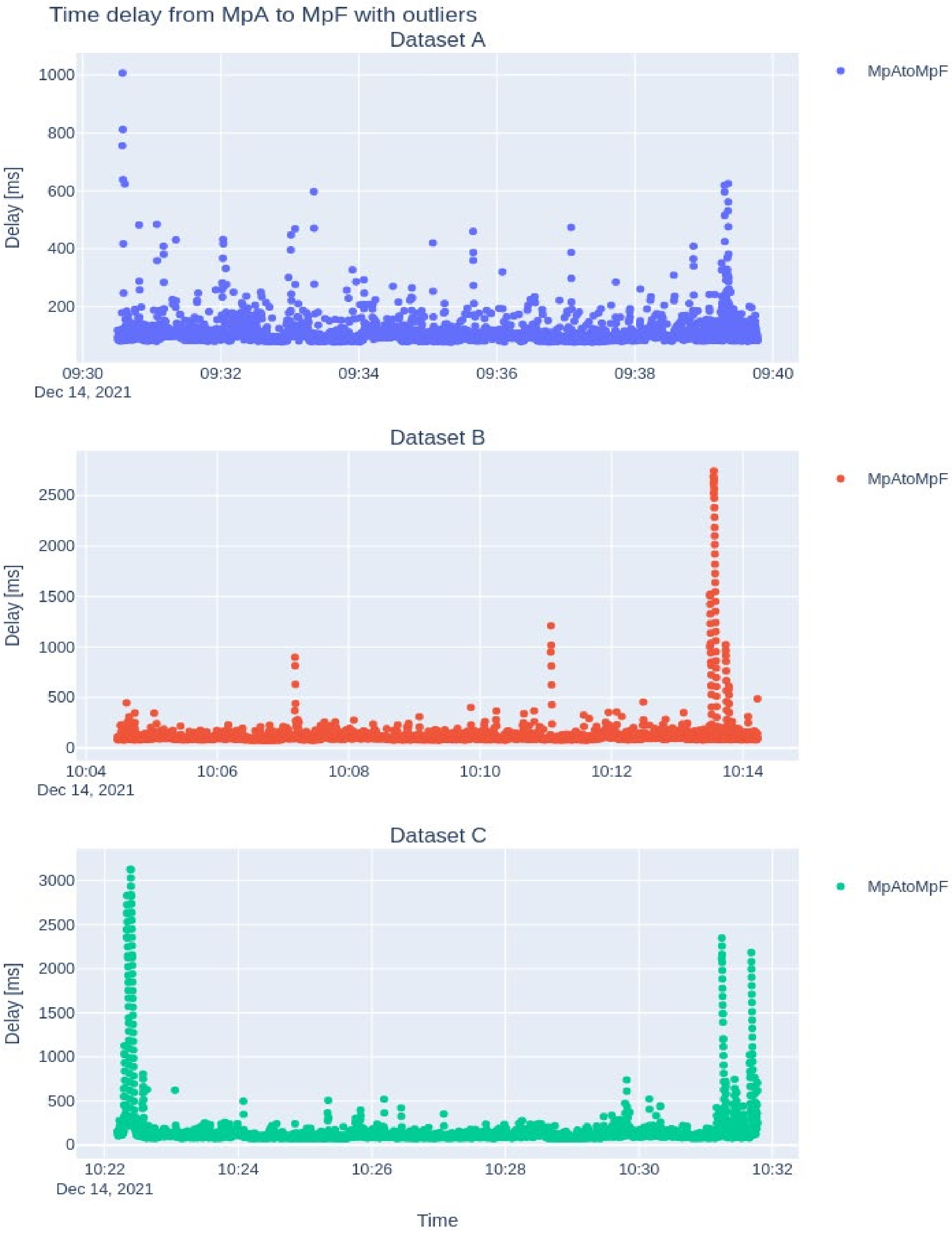


Figure 5-2 Round trip time from MpA to MpF with outliers

By filtering the data in Figure 5-2 for outliers, the spikes will be removed, and a more representative picture of the measurements will be given. Since the outliers are rather significant, they will be further examined in chapter 5.5. For detecting outliers the interquartile range method[76] is to be used as the dataset is rather skewed. The formula is shown in (5.1), where x is an adjustable factor for tuning the value for the outliers. In this project, $x = 3.5$ for all three data sets.

$$\begin{gathered} Q1 \; = \; 25\% \, percentile \\ Q3 \; = \; 75\% \, percentile \\ IQR \; = \; Q3 \; - \; Q1 \\ Outliers > Q3 + x * IQR \end{gathered} \tag{5.1}$$

After removing the outliers, it is possible to visually inspect the plot in Figure 5-3 to see that the spikes are gone.

In Table 5-2, the mean values for the round-trip times (MpAtoMpF), upload latency (MpAtoMpC), download latency (MpDtoMpF) and “backend-to-backend latency” (MpCtoMpD) are listed. For all three data sets, the download latency is within a range of 26 to 28 ms, while for the upload latency, data set A and B are the same and data set C is 8 ms higher. Generally, for all the datasets, the upload time is higher than the download time, even though the payload downloaded is higher than the one uploaded.

MpCtoMpD is the latency from a packet is sent from the backend service to the interchange and received again, it includes both the upload and download latency to the NWIXN and the latency within the NWIXN itself. For the three datasets, it ranges from 68 to 70 ms,

A simple ping test from the backend service to the interchange, gives a mean value of 31 ms. This gives that 62ms of the latency in MpCtoMpD comes from latency in the network, while the remaining is the latency within the interchange itself. As per the time this test was performed, the interchange wasn’t optimized for low latency[2] as it does some additional checks on the application properties[77].

Table 5-2 Mean value for round-trip, upload, download and “backend-to-backend”-latency

	MpAtoMpF	MpAtoMpC	MpDtoMpF	MpCtoMpD
Dataset A	104 ms	36 ms	27 ms	68 ms
Dataset B	106 ms	36 ms	26 ms	69 ms
Dataset C	112 ms	42 ms	28 ms	70 ms

[2] According to Christian Berg Skjetne, Senior engineer at NPRA, responsible for interchange development in Nordic Way 3

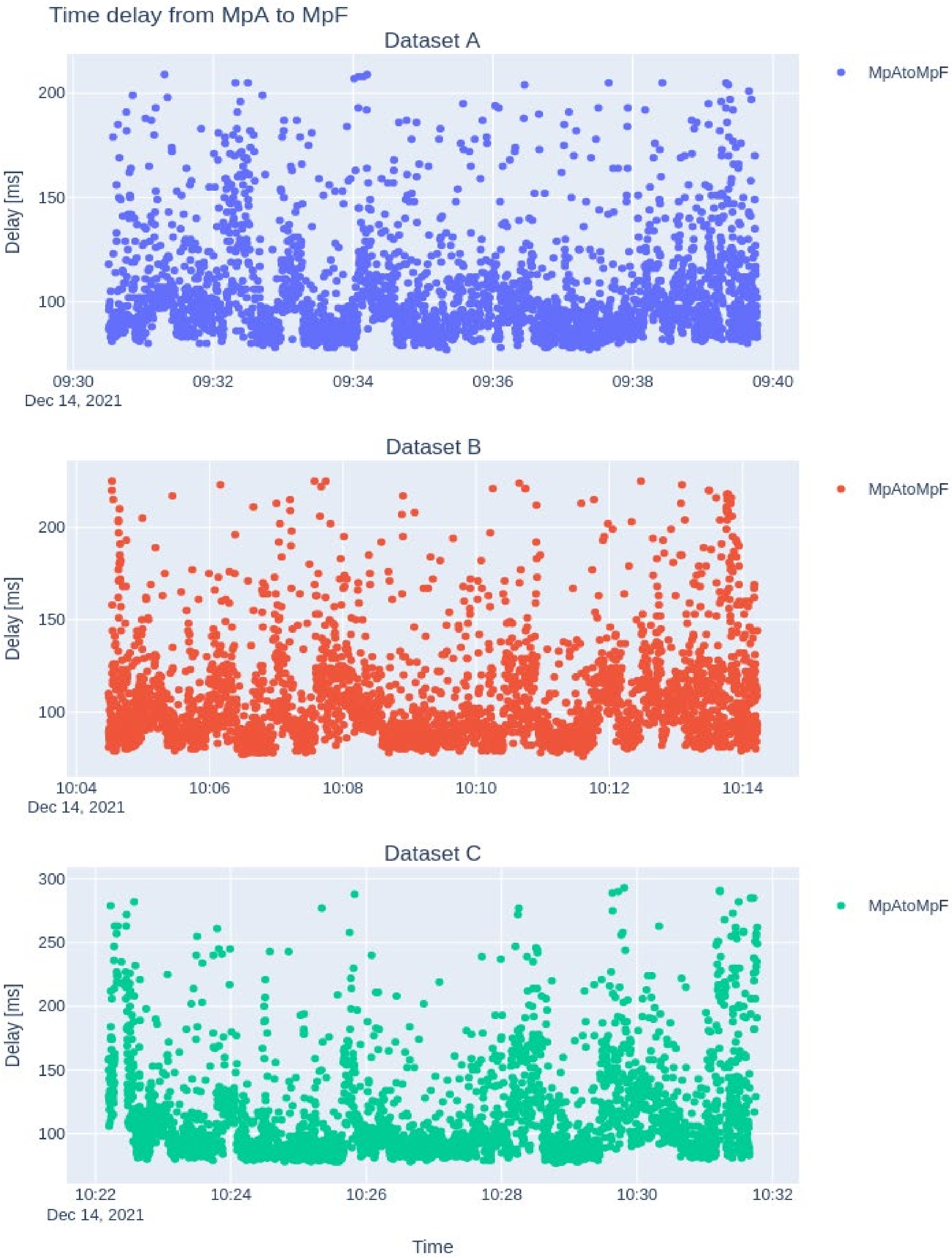


Figure 5-3 Round trip time from MpA to MpF without outliers

5.4 Correlation with signals from cellular network

An interesting observation in Figure 5-3, is the fields where the round trip-time gets an increase, for instance in dataset A, from time 09.34.00 to 09.34.30. This pattern occurs in all datasets. In Figure 5-6, Figure 5-5 and Figure 5-6, the round-trip time (MpAtoMpF) is plotted together with the number of Carrier Aggregation bands (CA-bands) and the network type (LTE/5G NSA). From this plot it is possible to see that whether it is LTE or 5G and how many CA-bands it is connected to has a correlation with the round-trip time.

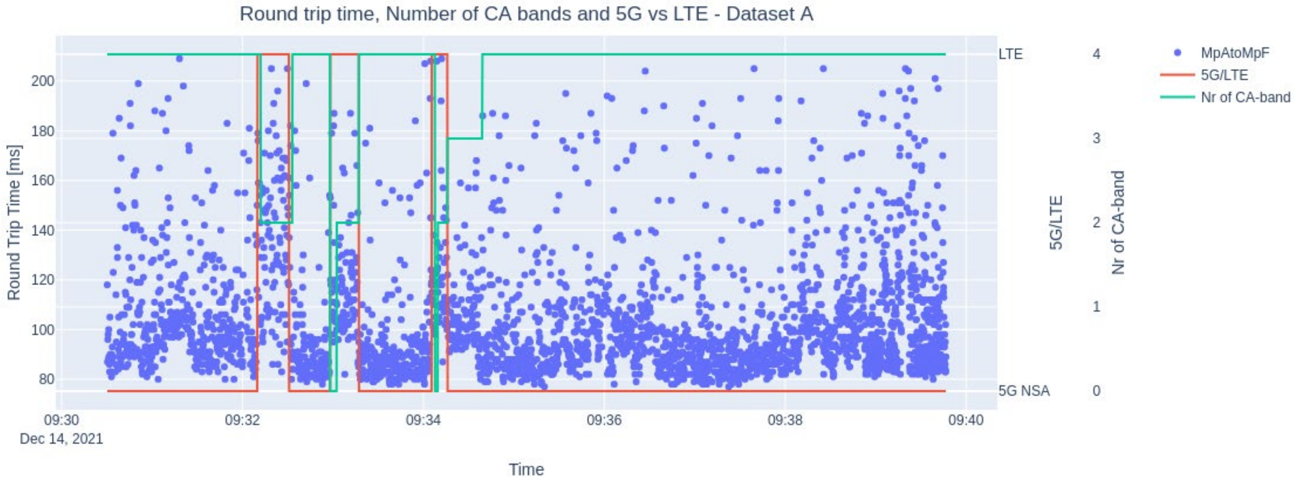


Figure 5-4 Round-trip time plotted together the number of CA bands and LTE/5G for dataset A

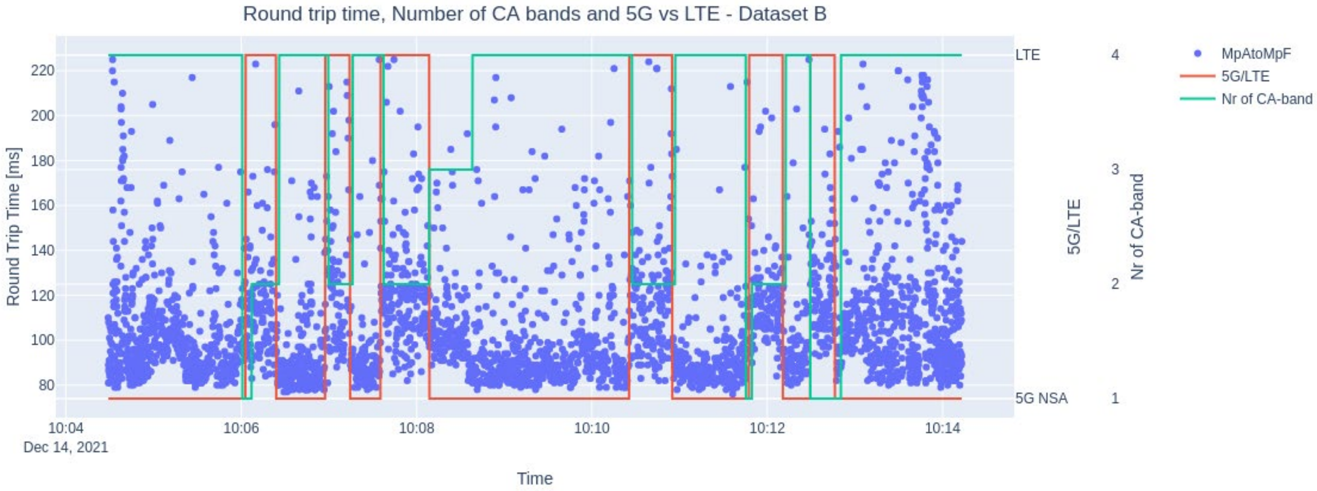


Figure 5-5 Round-trip time plotted together the number of CA bands and LTE/5G for dataset B

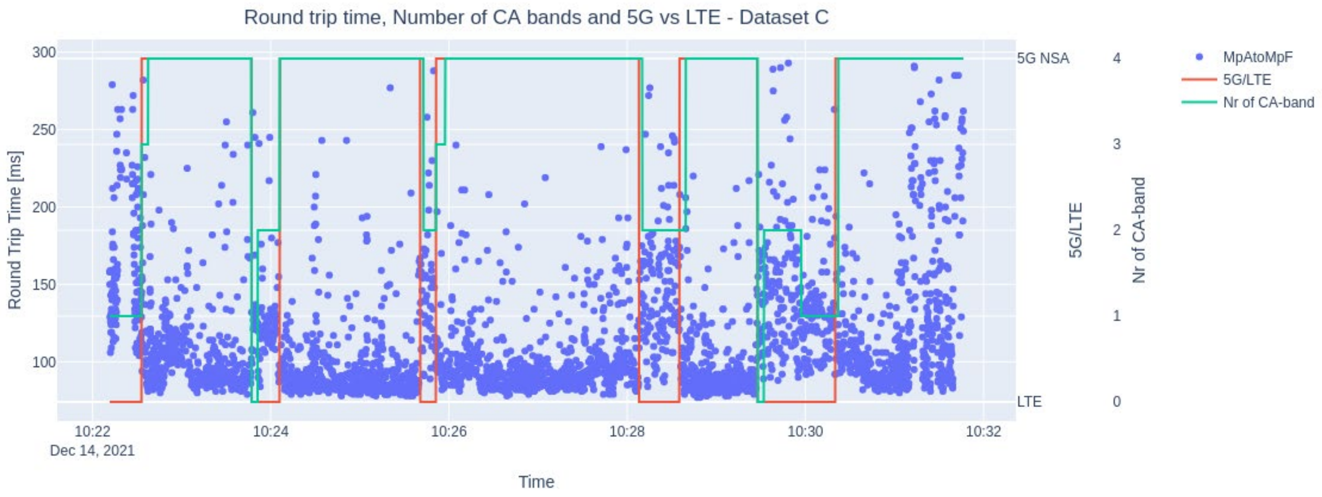
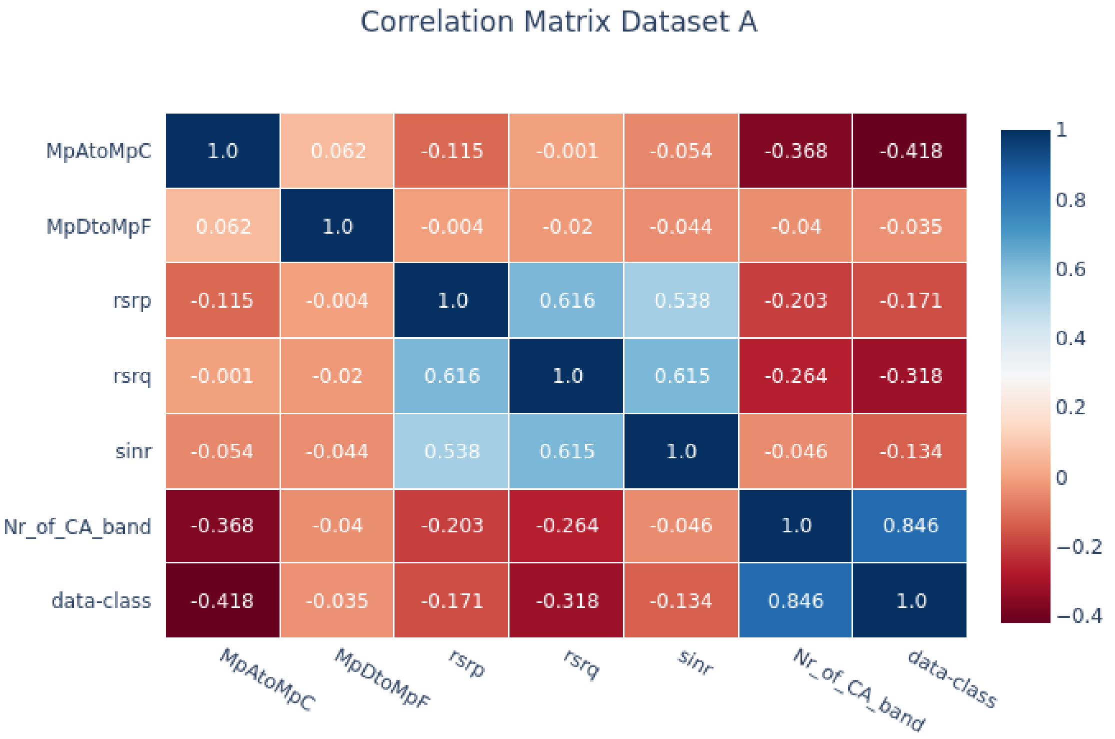


Figure 5-6 Round-trip time plotted together the number of CA bands and LTE/5G for dataset C

To check for correlation with the other parameters of interest that have been logged from the Mikrotik router, a correlation matrix for each data set have been made in Figure 5-9, Figure 5-8 and Figure 5-9. In these matrices, the upload latency (MpAtoMpC) and download latency (MpDtoMpF) are correlated with the cellular network signals. It is clear to see that the upload latency is in correlation with the number of CA-bands and network type. For the download latency, it shows no correlation with any of the cellular network signals.



	MpAtoMpC	MpDtoMpF	rsrp	rsrq	sinr	Nr_of_CA_band	data-class
MpAtoMpC	1.0	0.062	-0.115	-0.001	-0.054	-0.368	-0.418
MpDtoMpF	0.062	1.0	-0.004	-0.02	-0.044	-0.04	-0.035
rsrp	-0.115	-0.004	1.0	0.616	0.538	-0.203	-0.171
rsrq	-0.001	-0.02	0.616	1.0	0.615	-0.264	-0.318
sinr	-0.054	-0.044	0.538	0.615	1.0	-0.046	-0.134
Nr_of_CA_band	-0.368	-0.04	-0.203	-0.264	-0.046	1.0	0.846
data-class	-0.418	-0.035	-0.171	-0.318	-0.134	0.846	1.0

Figure 5-7 Correlation matrix between upload and download latency and cellular network signals for dataset A

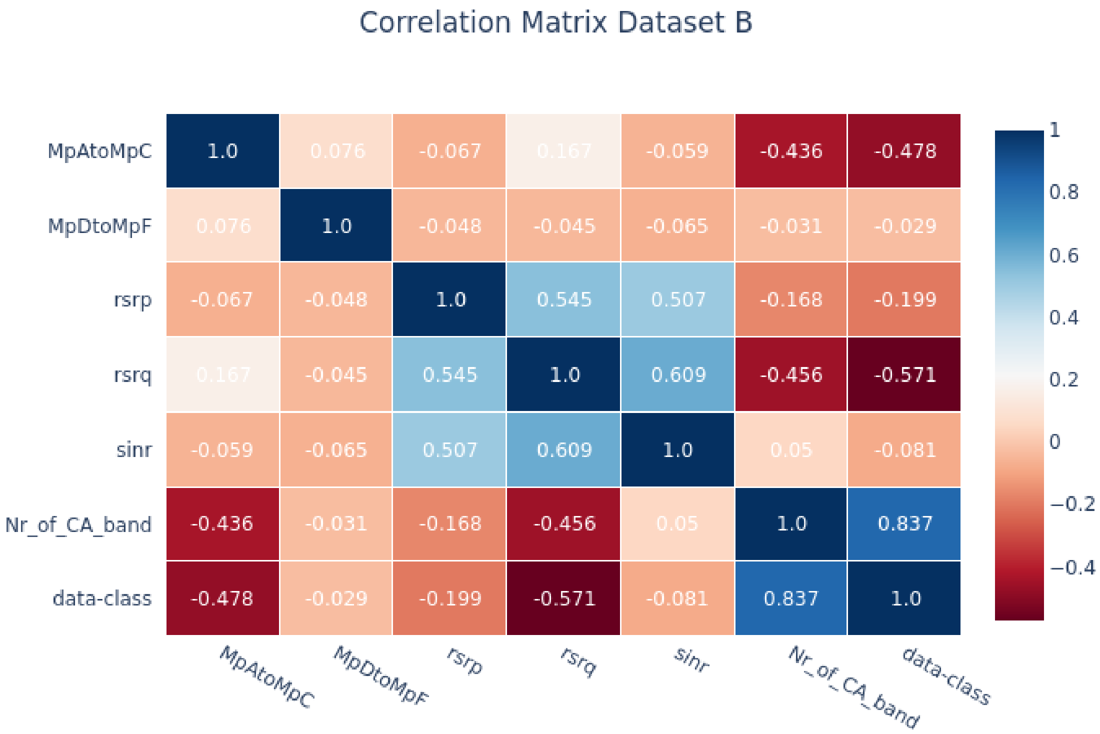


Figure 5-8 Correlation matrix between upload and download latency and cellular network signals for dataset B

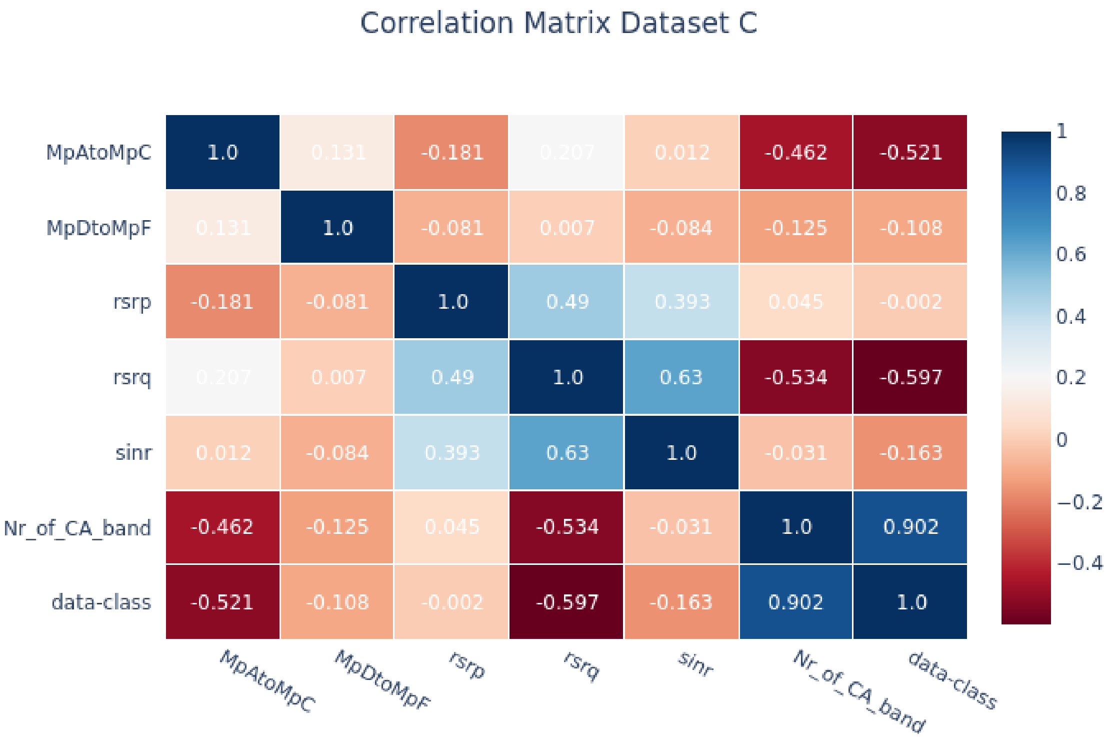


Figure 5-9 Correlation matrix between upload and download latency and cellular network signals for dataset C

5.5 Outliers

As some of the outliers where rather significant, it is of interest to have a closer look at them. In Figure 5-10, the time components in the outliers are plotted in bars. It is clear to see several staircase patterns in the plot. These staircase pattern occur on successive measurements and indicates that the packets are being buffered before being sent again. Looking closer on the elements in the bars, the highest stairs are composed of a latency in both the upload latency (MpAtoMpF) and the “backend-to-backend” latency (MpCtoMpD). This can indicate that there have been some issues around the backend service installed in the cloud and are not related to the cellular network.

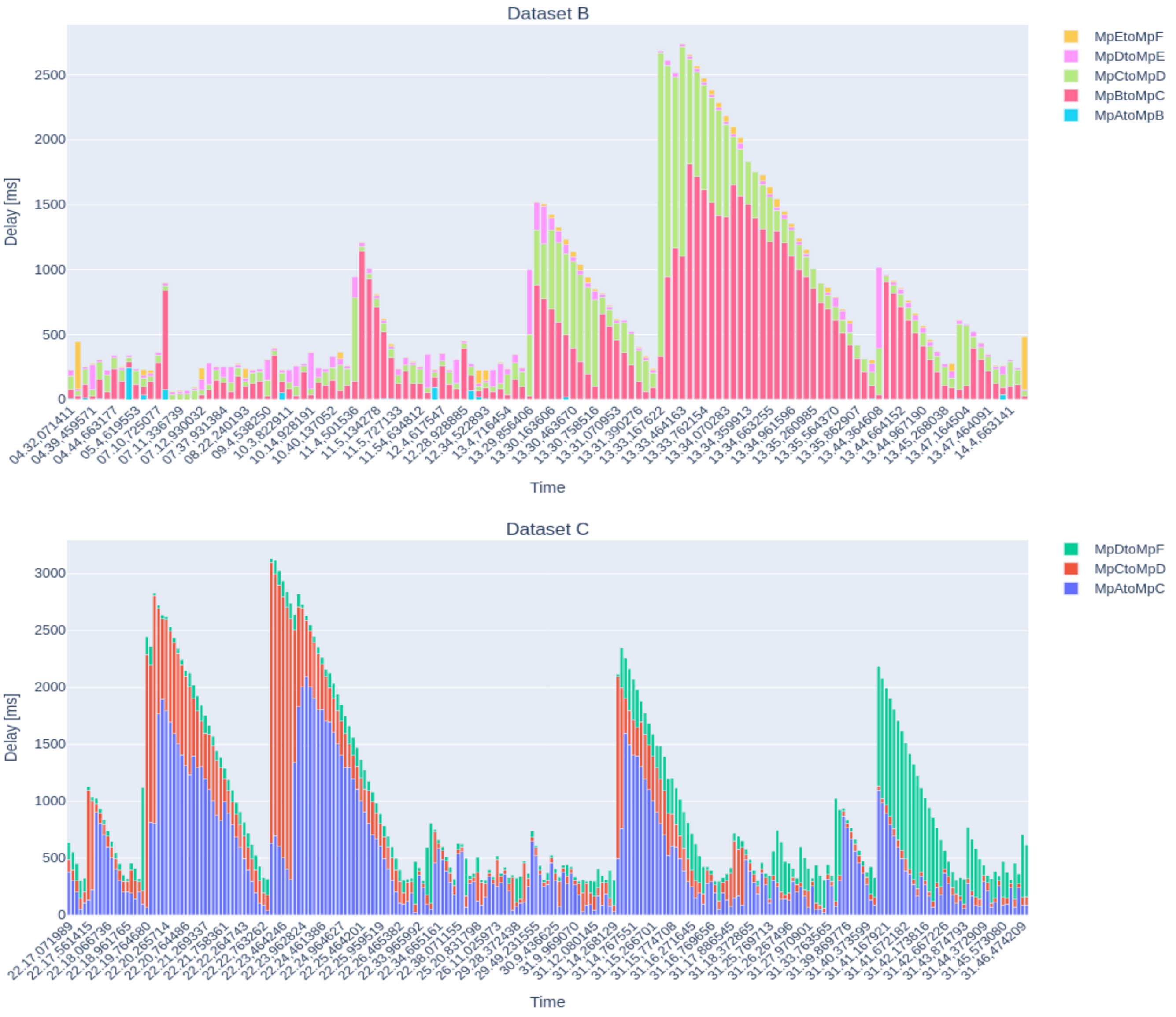


Figure 5-10 Time components in outliers for the dataset B and C

5.6 Driving route

From chapter 5.4, the plots indicate that the data-class isn’t concise through the three data sets. To visualize this, the data-class is plotted on a map in Figure 5-11, Figure 5-12 and

Figure 5-13. It is easy to see the difference in where 5G coverage is obtained on the same route over three different trials.

Another interesting observation is the loss of position fix, especially in Figure 5-11, there is a large section missing position fix which is marked by a red rectangle.

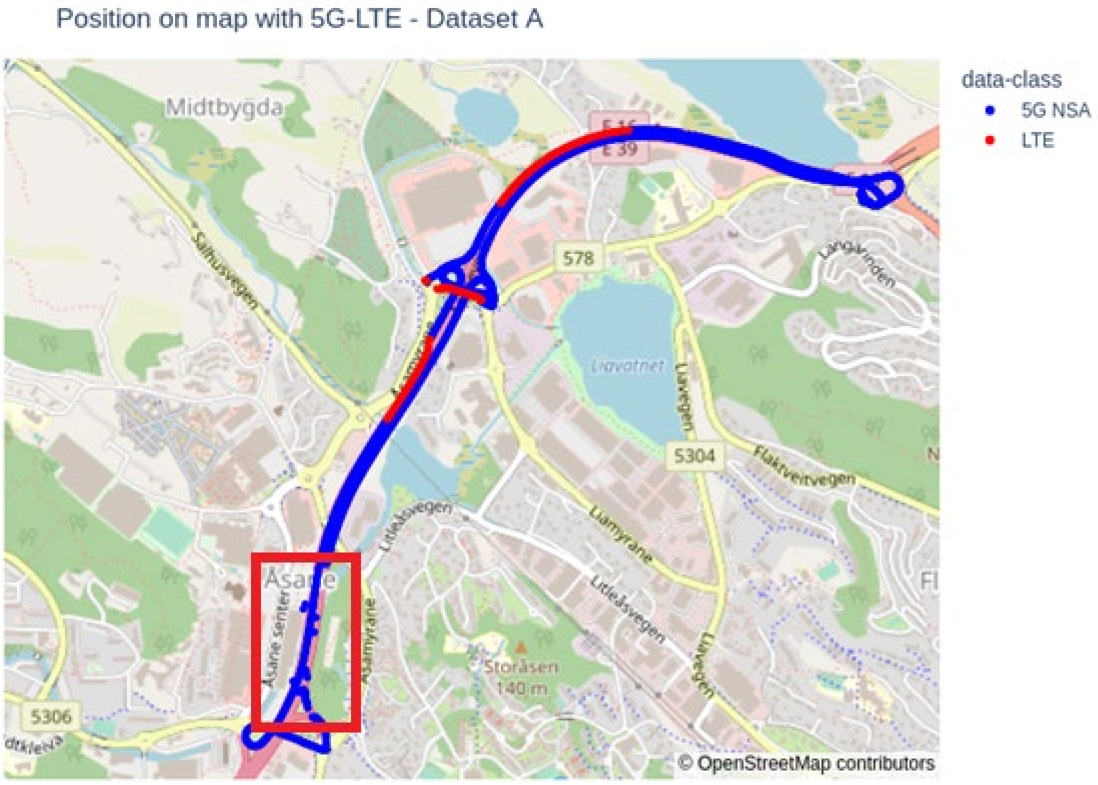


Figure 5-11 Collected positions from dataset A with different colored dots based on the data-class

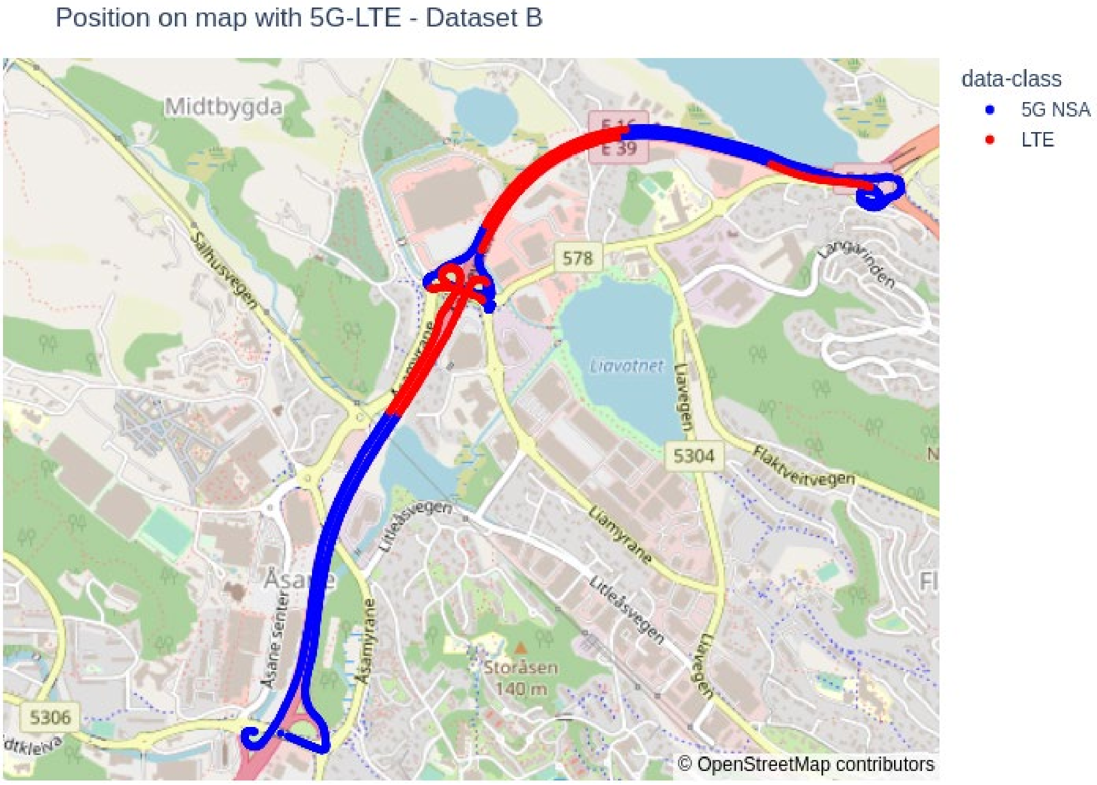


Figure 5-12 Collected positions from dataset B with different colored dots based on the data-class

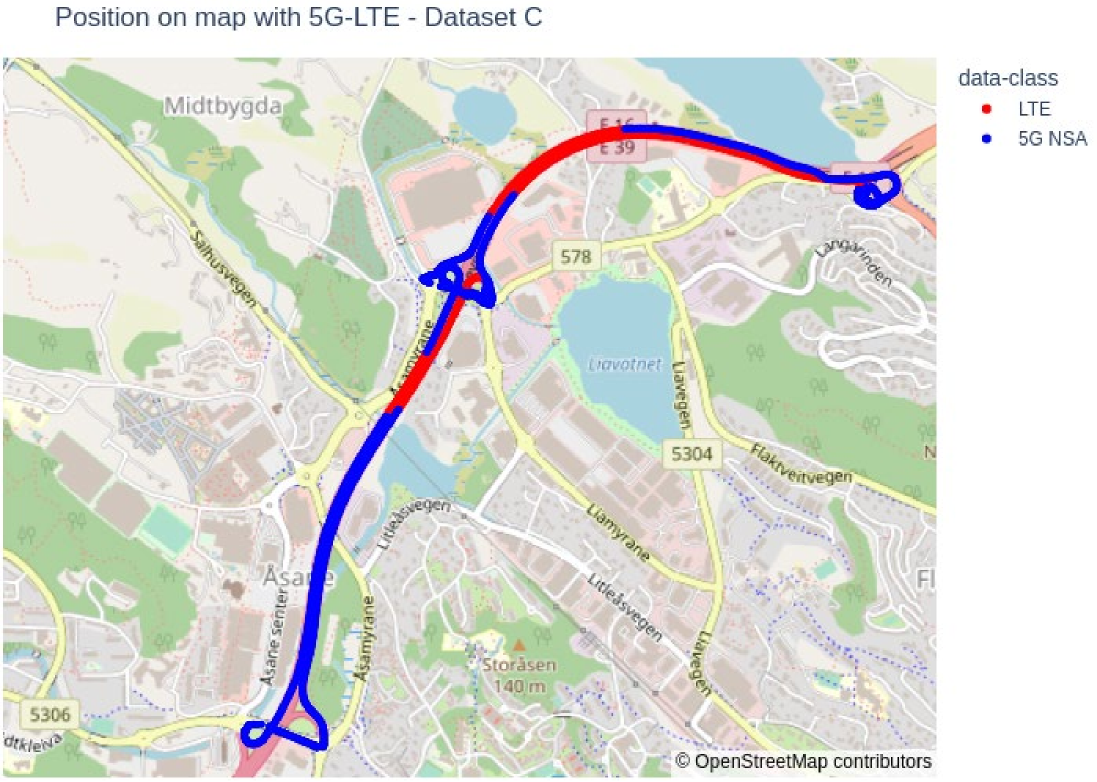


Figure 5-13 Collected positions from dataset C with different colored dots based on the data-class

6 Discussion

6.1 Use cases

Chapter 3.1 show an infrastructure based wrong way ahead system based on camera detection and warning of other drivers by lowering the speed limit and lane closure signs. The latency in the system is substantial and not deterministic as the vehicle moving wrong way needs to be detected by two cameras to trigger the automatic warning. The availability of the system relies on the camera system and that the cameras are maintained in a way that the traffic is visible for the camera.

In chapter 3.2, the C-Roads platform have specified a use case that includes a “human-in-the-loop” to trigger the warning of other vehicles. The same use case is defined to be used over hybrid communication and can be implemented towards e.g., the Nordic Way interchange and short-range communication (e.g., ITS-G5) according to the specification. This use case mostly specifies the information flow in a “Wrong way driver” scenario.

Chapter 3.3 concerns two standards from ETSI, both relying on C-ITS to detect wrong way drivers. The detection method is based on that all vehicles sends out CAM’s and that either the vehicles ITS-station, or a roadside ITS-station detects the wrong way vehicles and transmits a DENM to all nearby vehicles. The detection criteria are that the vehicles trajectory is in countersense direction. Among the requirements in the standards, the transmittal frequency of CAM shall be 10 Hz and one standard operate with a maximum of 100ms latency, while the other operates with 300ms latency.

This report has a real-life episode from Knappe tunnel in chapter 3.1.1 and this episode gives valuable insight about how an actual scenario takes place, and the episode raises an interesting question related to the standards in chapter 3.3. As the wrong way driving car in Figure 3-3 is passing in front of the emergency braking car, the trajectory is about 90 degrees on to the traffic direction. The maneuver the car does, is something that could be normal in a traffic jam situation where it would sneak into the line. Raising a wrong way ahead warning for this would then reduce the trust in the system. If the episode from Knappe tunnel triggered a warning when the vehicle was in a countersense direction, it still wouldn’t be able to warn the car that does the emergency braking in Figure 3-3.

Another interesting question from the Knappe tunnel episode is, what if it was a vehicle in the left lane? This question concerns the human aspect of driving, and generally, the closer the vehicles are to each other, the more one must rely on the reaction of the drivers to stop. So, it is important to have in mind that the technology today, can only help us as far as the human driver can consume the given information and act on it in predictable manner.

6.2 Latency

From chapter 5.3, where latency of the data has been analyzed, we have a mean value from 104 ms to 112 ms for the round-trip time when outliers are removed. Comparing this to the first use case presented by ETSI in chapter 3.3 which has a latency requirement of 100 ms, this is clearly outside of the requirements. For the second ETSI use case, which have a 300 ms general requirement for collision detection, the results are within, even when considering that the maximum value in Figure 5-3 close to 300ms. Comparing again with the C-Roads use case in chapter 3.2, and the infrastructure-based system and episode from Knappe tunnel in chapter 3.1, these results are well within what those can perform in terms of latency, even if we count in the outliers which maxed out on about 3 seconds.

The difference in upload and download latency in Table 5-2, is about 10 ms higher for upload latency. This is then higher than the difference in upload and download latency in chapter 2.4. The measurements done in this project doesn't hold data that can help conclude on why this happens, and it can also be discussed if this is of interest at all, as one of the key features that is to be implemented in 5G is the ultra-low latency.

Another part of the round-trip time latency is the latency in the "backend-to-backend" communication that involves the Nordic Way interchange. Most of this latency originates from latency in the network and little in the NWIXN itself. One parameter for deploying a backend service could then be to have a network-wise closeness to the NWIXN to reduce the latency in the network.

The NWIXN-architecture described in chapter 2.3, supports node-to-node communication, meaning that the messages might need to travel through two NWIXN-nodes. It adds another link to the chain, that potentially can impact the reliability and latency of such a system. In this project, it has only been tested with one node and thus without node-to-node communication, and an interesting test would be to include more nodes in the test. An important aspect of a test of this kind, is how foreign vehicles that are connected to backend services and interchange nodes that are abroad is functioning in different use-cases.

To reduce latency on the backend side, one approach can also be to deploy NWIXN-nodes and backend services closer to the "edge", the exchange of information will then happen closer to where the vehicles travel, opposed to the backend service in 4.3 that is deployed in Finland which leads to longer network latency. With the II-interface in the Nordic Way architecture in chapter 2.3, this enables the interchange nodes to be deployed close to the edge as each new node posts its capabilities to other nodes and can probably with further development be automated in deploying on the edge. To select which interchange nodes and backend services a vehicle shall use, a similar test method as used in this project can probably be used to measure latency from a vehicle and through several backend systems and interchange nodes to select the one with lowest latency.

6.3 Cellular

From chapter 5.4, it is possible to see that the amount of carrier aggregation bands and if the data-class is 5G or LTE are correlated with the upload time of the data, while for download it didn't correlate. This is an important discovery, as these parameters can help decide what services are possible to deliver in certain areas. It is therefore also important to know in advance where one could expect to acquire these prerequisites. In chapter 5.6, it is possible to see that the same route gives different areas where LTE and 5G is available. This makes it hard to predict when one would achieve potential prerequisites for different use cases.

It would have been suspected that the cellular network signals, (RSRP, RSRQ and SINR) would have an impact on the results, but the correlation matrix from chapter 5.4 shows little to no correlation between the upload or download time and the signal parameters. This test is done with relatively small packets, and larger packet size would possibly have affected the results in the correlation matrix.

6.4 Positioning

The positioning of vehicles needs to be better than what have been presented in chapter 5.6, as it on some occasions missed many positions and drifted outside the road. Positioning is an important part of the CAM in chapter 2.2.2, which is the message that needs to be transmitted from each vehicle. For a use case like this, it is of importance to have confidence in the position to ensure that the vehicle is on the right side of the road barrier or isn’t map matched onto parallel, over or under lying roads. The results could have been better by utilizing dead reckoning, odometer, RTK and sensor fusion among these. This functionality is already available on the GNSS-receiver used in chapter 4.2.4 and dead reckoning and odometer should be rather trivial to implement on new vehicles to improve the positioning

For enabling RTK, this will need operational base stations that can supply correctional messages to the GNSS-receiver, and these stations will need to be installed, operated, and maintained to supply correctional messages reliably. Accurate positioning will be of importance within many use cases which enables C-ITS.

6.5 Future work

The future work that is of most importance is to test with a federated interchange network, and further how this work with foreign vehicles entering other countries. This will give valuable insight in how one could develop a deployment model, both for the deployment of interchange nodes and for backend services. Within this deployment model, it needs to be considered how close to the edge it needs to be deployed, and the model needs to be evaluated based on the services and use cases it shall support.

For future testing that involves cellular network, the use of more professional grade equipment will help to identify when UE switches cells, identify handover delays and identify which base stations are used. The deployment of 5G-SA comes with possibility of network slicing and low latency, and investigating these features is of interest within the C-ITS field.

In chapter 5.5, a brief analysis of the outliers was performed where it was indicated that the delay with staircase patterns occurred near the backend solution, it would be of interest to investigate these outliers further to figure out how to avoid them.

Improvement in the positioning, and especially positioning with RTK for moving vehicles will be of importance in the future. Future work should be done in distribution of correctional messages in a standardized way.

7 Conclusion

The research objective in this project was to look at the possibility to use C-ITS instead of infrastructure-based sensor systems to warn other drivers if a wrong way driver is approaching, by utilizing the Nordic Way interchange architecture and cellular communication.

From the test that was done with a test system, the mean round-trip latency from a message was sent from a vehicle, to it was received again was 104 ms to 112 ms. The results also show a correlation between the upload time and the amount of carrier aggregation bands available and if it was connected to a 5G network. These findings demonstrates that it is possible for vehicles to send messages, utilizing the Nordic Way architecture, to warn other drivers about wrong way drivers.

This is limited to vehicles that are C-ITS enabled and transmits cooperative awareness messages to a backend service that have functionality for detecting wrong way drivers implemented. The backend service can then transmit a warning to the Nordic Way interchange for others to retrieve and further warn.

It is of importance for road operators to start facilitating for the exchange of messages between actors within the C-ITS field, as this enables the establishment of use cases that can replace infrastructure-based systems that are costly to install and maintain.

8 References

[1] S. Simpson and D. Bruggeman, "Detection and Warning Systems for Wrong-Way Driving," (in English), Tech Report 2015. [Online]. Available: https://rosap.ntl.bts.gov/view/dot/29573.

[2] Statens vegvesen. "Søk i Trafikkulykkesregisteret." https://trine.atlas.vegvesen.no/ (accessed 19.11, 2021).

[3] R. Ramsdal. "Stålpigger skal stoppe sløve bilister." https://www.tu.no/artikler/stalpigger-skal-stoppe-slove-bilister/236795 (accessed 12.01, 2022).

[4] *Commission Delegated Regulation (EU) No 886/2013 of 15 May 2013 supplementing Directive 2010/40/EU of the European Parliament and of the Council with regard to data and procedures for the provision, where possible, of road safety-related minimum universal traffic information free of charge to users Text with EEA relevance*, 2013.

[5] Cambridge Dictionary "dual carriageway." https://dictionary.cambridge.org/dictionary/english/dual-carriageway (accessed 15.08.2021.

[6] *Håndbok N100 Veg-og gateutforming*, N100, S. vegvesen, 2021. [Online]. Available: https://svv-cm-sv-apppublic-prod.azurewebsites.net/product/859921/nb

[7] *Intelligent Transport Systems (ITS); Communications Architecture*, ETSI EN 302 665 V1.1.1 (2010-09), European Telecommunications Standards Institute, 2010. [Online]. Available: https://www.etsi.org/deliver/etsi_en/302600_302699/302665/01.01.01_60/en_302665v010101p.pdf

[8] Thales. "Hardware Security Modules (HSMs)." https://cpl.thalesgroup.com/encryption/hardware-security-modules (accessed 26.9, 2021).

[9] *Intelligent Transport Systems (ITS); Vehicular Communications; Basic Set of Applications; Facilities layer protocols and communication requirements for infrastructure services; Release 2*, ETSI TS 103 301 V2.1.1 (2021-03), European Telecommunications Standards Institute, 2021. [Online]. Available: https://www.etsi.org/deliver/etsi_ts/103300_103399/103301/02.01.01_60/ts_103301v020101p.pdf

[10] European Telecommunications Standards Institute, "Intelligent Transport Systems (ITS); Security; Pre-standardization Study on ITS Facility Layer Security for C-ITS Communication Using Cellular Uu Interface," ETSI.org, Tech. Rep ETSI TR 103 630 V1.1.1 (2020-11), 2020. [Online]. Available: https://www.etsi.org/deliver/etsi_tr/103600_103699/103630/01.01.01_60/tr_103630v010101p.pdf

[11] *Intelligent Transport Systems (ITS); Vehicular Communications; Basic Set of Applications; Part 2: Specification of Cooperative Awareness Basic Service*, ETSI

EN 302 637-2 V1.4.1 (2019-04), E. T. S. Institute, 2019. [Online]. Available: https://www.etsi.org/deliver/etsi_en/302600_302699/30263702/01.04.01_60/en_30263702v010401p.pdf

[12] *Intelligent Transport Systems (ITS); Users and applications requirements; Part 2: Applications and facilities layer common data dictionary*, ETSI TS 102 894-2 V1.3.1 (2018-08), E. T. S. Institute, 2018. [Online]. Available: https://www.etsi.org/deliver/etsi_ts/102800_102899/10289402/01.03.01_60/ts_10289402v010301p.pdf

[13] *Intelligent Transport Systems (ITS); Vehicular Communications; Basic Set of Applications; Part 3: Specifications of Decentralized Environmental Notification Basic Service*, ETSI EN 302 637-3 V1.3.1 (2019-04), E. T. S. Institute, 2019. [Online]. Available: https://www.etsi.org/deliver/etsi_en/302600_302699/30263703/01.03.01_60/en_30263703v010301p.pdf

[14] *C-ITS IP Based Interface Profile R1.8.0*, C-Roads, 2020.

[15] J. Wannstrom. "Carrier Aggregation explained." 3GPP. https://www.3gpp.org/technologies/keywords-acronyms/101-carrier-aggregation-explained (accessed 1.10, 2021).

[16] M. Kottkamp, A. Pandey, A. Roessler, R. Stuhlfauth, and D. Raddino, *5G New Radio - Fundamentals, Procedures, Testing Aspects*, 22 ed. Rohde & Schwarz: Rohde & Schwarz GmbH & KG, 2019.

[17] *3rd Generation Partnership Project; Technical Specification Group Radio Access Network; Evolved Universal Terrestrial Radio Access (E-UTRA); Study on latency reduction techniques for LTE (Release 14)*, 3GPP TR 36.881 V14.0.0 (2016-06), 3GPP, 2016. [Online]. Available: https://portal.3gpp.org/desktopmodules/Specifications/SpecificationDetails.aspx?specificationId=2901

[18] L. C. Gimenez, M. C. Cascino, M. Stefan, K. I. Pedersen, and A. F. Cattoni, "Mobility Performance in Slow- and High-Speed LTE Real Scenarios," in *2016 IEEE 83rd Vehicular Technology Conference (VTC Spring)*, 15-18 May 2016 2016, pp. 1-5, doi: 10.1109/VTCSpring.2016.7504347.

[19] J. Lozada. "RSRP, RSRQ and SINR – LTE vs 5G." https://redesmoviles.com/5g/rsrp-rsrq-sinr/ (accessed 28.12, 2021).

[20] *Directive 2004/54/EC of the European Parliament and of the Council of 29 April 2004 on minimum safety requirements for tunnels in the Trans-European Road Network*, 2004.

[21] *Forskrift om minimum sikkerhetskrav til visse vegtunneler (tunnelsikkerhetsforskriften)*, 2007.

[22] *Common C-ITS Service and Use Case Definitions Version 2.0.0*, C-Roads, 2021.

[23] *Intelligent Transport Systems (ITS); Vehicular Communications; Basic Set of Applications; Release 2*, ETSI TR 102 638 V1.1.1 (2009-06), European Telecommunications Standards Institute, 2009. [Online]. Available:

https://www.etsi.org/deliver/etsi_tr/102600_102699/102638/01.01.01_60/tr_102638v010101p.pdf

[24] *Intelligent Transport Systems (ITS); V2X Applications; Part 1: Road Hazard Signalling (RHS) application requirements specification*, ETSI TS 101 539-1 V1.1.1 (2013-08), European Telecommunications Standards Institute, 2013. [Online]. Available: https://www.etsi.org/deliver/etsi_ts/101500_101599/10153901/01.01.01_60/ts_10153901v010101p.pdf

[25] *Intelligent Transport Systems (ITS); V2X Applications; Part 2: Intersection Collision Risk Warning (ICRW) application requirements specification*, ETSI TS 101 539-2 V1.1.1 (2018-06), European Telecommunications Standards Institute, 2018. [Online]. Available: https://www.etsi.org/deliver/etsi_ts/101500_101599/10153902/01.01.01_60/ts_10153902v010101p.pdf

[26] *Intelligent Transport Systems (ITS); V2X Applications; Part 3: Longitudinal Collision Risk Warning (LCRW) application requirements specification*, ETSI TS 101 539-3 V1.1.1 (2013-11), European Telecommunications Standards Institute, 2013. [Online]. Available: https://www.etsi.org/deliver/etsi_ts/101500_101599/10153903/01.01.01_60/ts_10153903v010101p.pdf

[27] B. M. Elnes, "Project report: Data from C-ITS server to DATEX-II v3.0 server," Aventi Intelligent Communication, Technical Report 28.05.2020, 2020.

[28] M. Boban, A. Kousaridas, K. Manolakis, J. Eichinger, and W. Xu, "Connected Roads of the Future: Use Cases, Requirements, and Design Considerations for Vehicle-to-Everything Communications," *IEEE Vehicular Technology Magazine,* vol. 13, no. 3, pp. 110-123, 2018, doi: 10.1109/mvt.2017.2777259.

[29] P. Arnesen *et al.*, "LambdaRoad - Summarizing the main findings of work package 1: System and organizational requirements for CCAM"," 2020. [Online]. Available: https://hdl.handle.net/11250/2677779

[30] T. Ojanperä, J. Mäkelä, O. Mämmelä, M. Majanen, and O. Martikainen, "Use Cases and Communications Architecture for 5G-Enabled Road Safety Services," in *2018 European Conference on Networks and Communications (EuCNC)*, 18-21 June 2018 2018, pp. 335-340, doi: 10.1109/EuCNC.2018.8443193.

[31] A. Svindseth, "Test system for testing ITS-services over 4G cellular network " University of Southeast-Norway, Tech Rep. 2020. [Online]. Available: https://www.halvorsen.blog/documents/projects/student_projects/master/2020/Statens%20vegvesen/Project%20Report_Statensvegvesen_2020.pdf

[32] "Regions and zones." https://cloud.google.com/compute/docs/regions-zones (accessed 3.12, 2021).

[33] A. Svindseth. "msc_project_utilities." https://github.com/svinz/msc_project_utilities (accessed 11.01, 2022).

[34] "What is Elasticsearch?" https://www.elastic.co/guide/en/elasticsearch/reference/current/elasticsearch-intro.html (accessed 26.09, 2020).

[35] "NGINX Ingress Controller." https://kubernetes.github.io/ingress-nginx/ (accessed 11.01, 2022).

[36] "Eclipse Mosquitto." https://github.com/eclipse/mosquitto (accessed 26.09, 2020).

[37] A. Svindseth. "msc_project_onboardunit." https://github.com/svinz/msc_project_onboardunit (accessed 11.01, 2022).

[38] aiohttp. "Welcome to AIOHTTP." https://docs.aiohttp.org/en/stable/ (accessed 15.08.2021.

[39] K. Reitz. "Requests: HTTP for Humans™." https://requests.readthedocs.io/en/master/ (accessed 1.11, 2020).

[40] gpsd. "gpsd — a GPS service daemon." https://gpsd.gitlab.io/gpsd/index.html (accessed 15.08, 2021).

[41] S. Finer. "Welcome to sslkeylog's documentation!" https://sslkeylog.readthedocs.io/en/latest/ (accessed 14.08, 2021).

[42] L. Root. "asyncio: We Did It Wrong." https://www.roguelynn.com/words/asyncio-we-did-it-wrong/ (accessed 07.08, 2021).

[43] Mikrotik. "Mikrotik Chateau 5G." https://mikrotik.com/product/chateau_5g (accessed 29.08, 2021).

[44] G. E. Miller and E. S. Raymond. "GPSD Time Service HOWTO." https://gpsd.gitlab.io/gpsd/gpsd-time-service-howto.html#_1pps_quality_issues (accessed 15.08., 2021).

[45] Waveshare.com. "MAX-M8Q GNSS HAT for Raspberry Pi, Multi-constellation Receiver Support, GPS, Beidou, Galileo, GLONASS." https://www.waveshare.com/product/raspberry-pi/hats/max-m8q-gnss-hat.htm (accessed 15.08, 2021).

[46] *Product Summary MAX-M8 series*, UBX-16008997 - R06, u-blox, 2018.

[47] *u-blox 8 / u-blox M8 Receiver description Including protocol specification*, UBX-13003221 - R24, u-blox, 2021.

[48] "LinuxPPS wiki." http://linuxpps.org/doku.php/start (accessed 10.10, 2021).

[49] chrony. "Introduction." https://chrony.tuxfamily.org/index.html (accessed 15.08, 2021).

[50] P. O'Keeffe. "Rasbian NTP Server." https://github.com/patricktokeeffe/rpi-ntp-server/blob/master/readme.md (accessed 14.08, 2021).

[51] Google. "Configure NTP for your instances." https://cloud.google.com/compute/docs/instances/managing-instances#configure_ntp_for_your_instances (accessed 28.09, 2020).

[52] K. Bikos and A. Buckle. "What Is a Leap Second?" https://www.timeanddate.com/time/leapseconds.html (accessed 19.8, 2021).

[53] SparkFun. "SparkFun GPS-RTK Dead Reckoning pHAT for Raspberry Pi." https://www.sparkfun.com/products/16475 (accessed 19.9, 2021).

[54] u-blox, "ZED-F9R-01B u-blox F9 high precision sensor fusion GNSS receiver Data sheet," 2021. Accessed: 23.09.2021. [Online]. Available: https://www.u-blox.com/en/ubx-viewer/view/ZED-F9R-01B_Datasheet_UBX-19054459?url=https%3A%2F%2Fwww.u-blox.com%2Fsites%2Fdefault%2Ffiles%2FZED-F9R-01B_Datasheet_UBX-19054459.pdf

[55] F. v. Diggelen. "GPS accuracy: Lies, damn lies and statistics." https://www.gpsworld.com/gps-accuracy-lies-damn-lies-and-statistics/ (accessed 27.9, 2021).

[56] N. Seidle. "What is GPS RTK?" https://learn.sparkfun.com/tutorials/what-is-gps-rtk/all (accessed 09.10, 2021).

[57] *Intelligent Transport Systems (ITS); Facilities Layer function; Part 2: Position and Time management (PoTi); Release 2*, ETSI EN 302 890-2 V2.1.1 (2020-03), European Telecommunications Standards Institute, 2020.

[58] F. Gunnarsson and S. M. Razavi. "LTE Positioning and RTK: Precision down to the centimeter." https://www.ericsson.com/en/blog/2018/11/lte-positioning-and-rtk-precision-down-to-the-centimeter (accessed 9.10, 2021).

[59] tshark. "tshark.dev." https://tshark.dev/ (accessed 14.08, 2021).

[60] Wireshark. "About Wireshark." https://www.wireshark.org/ (accessed 14.08, 2021).

[61] K. Dargel, T. Lindhorst, H. Sychla, and wfailla. "TSHARK in a container." https://github.com/travelping/docker-pcap (accessed 15.08, 2021).

[62] A. Svindseth. "msc_project_tshark." https://github.com/svinz/msc_project_tshark (accessed 11.01, 2022).

[63] A. Svindseth. "msc_project_log_preprocessing_and_upload." https://github.com/svinz/msc_project_log_preprocessing_and_upload (accessed 11.01, 2022).

[64] G. Sim. "rhea - A reactive library for the AMQP protocol, for easy development of both clients and servers." https://www.npmjs.com/package/rhea (accessed 24.12, 2021).

[65] Y. Maguire. "mqtt - A library for the MQTT protocol." https://www.npmjs.com/package/mqtt (accessed 24.12, 2021).

[66] D. M. Clements, M. Colina, J. Sumners, and T. Watson. "pino - Very low overhead Node.js logger." https://www.npmjs.com/package/pino (accessed 24.12, 2021).

[67] A. Svindseth. "msc_project_backend_nodejs." https://github.com/svinz/msc_project_backend_nodejs (accessed 11.01, 2022).

[68] E. Gundersen, J. Sætre, and M. C. Stjernberg, "Ulykke i Åsane: Bil i feil retning på motorveien," in *Bergensavisen*, ed. ba.no, 2021.

[69] T. Opheim, "Bil kolliderte etter å ha kjørt feil vei i Åsane," in *Bergens Tidende*, ed, 2021.

[70] "Dekningskart." https://www.telia.no/nett/dekning/ (accessed 30.9, 2021).

[71] "Visual Studio Code." https://code.visualstudio.com/ (accessed 20.11, 2021).

[72] "Jupyter." https://jupyter.org/ (accessed 20.11, 2021).

[73] "pandas." https://pandas.pydata.org/ (accessed 20.11, 2021).

[74] "Plotly Python Open Source Graphing Library." https://plotly.com/python/ (accessed 20.11, 2021).

[75] A. Svindseth. "msc_project_analyses " https://github.com/svinz/msc_project_analyses (accessed 11.01, 2022).

[76] T. Courtney. "What Is the Interquartile Range Rule?" https://www.thoughtco.com/what-is-the-interquartile-range-rule-3126244 (accessed 21.12, 2021).

[77] C. B. Skjetne, "Conversation with Christian on Teams," 15.12.2021, 2021.

9 Appendices

Appendix A Original task description

FMH606 Master's Thesis

Title: Evaluation of services suitable to deliver over NordicWay interchange node
USN supervisor: Hans-Petter Halvorsen

External partner: Statens Vegvesen / Norwegian Public Road Administration (NPRA)

Task background:
The road vehicles of today are trending towards becoming more and more cooperative, connected and automated. To be able to achieve this, the vehicles needs to rely on more than sensors installed in the vehicle .
A key component in the vehicle of the future is communication from vehicle to vehicle (V2V) and vehicle to infrastructure (V2I). The V2I services can typical be “Geo fence”, “Road Works Warning”, “Traffic Ahead Warning”. As a road operator and road traffic authority, Statens vegvesen needs to start planning for implementing these services, and as a part of this, Statens Vegvesen participates in the NordicWay3 project. This project is a collaboration between Finland, Sweden, Denmark and Norway, and is partly financed by the Innovation and Networks Executive Agency (INEA) through the Connecting Europe Facility (CEF) program.

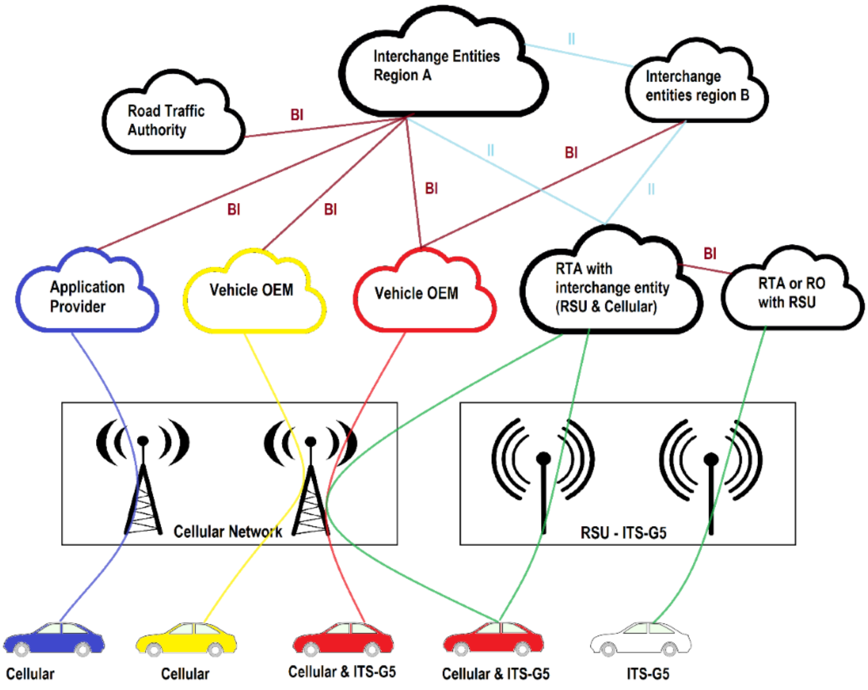


Figure 1 Communication between cars and backend

As a basis for some of these services, there is a need for a backend system that performs the interchange of this information, as have been visualized in Figure 1. In the earlier NordicWay projects, a prototype of the interchange node has been developed and used by the pilots in the projects. This interchange node will now be further developed in NordicWay3 and be brought closed to production and pilots in the NordicWay project will use the interchange node for testing. Since the interchange node will be a cloud solution and, in some cases rely on OEM clouds and cellular networks, it won't be a true hard real time system. An evaluation is therefore wanted of selected services to find out if they can be relevant for use together with the interchange node. Services are to be delivered wirelessly to the testunit in the vehicle.

Task description:

- Perform a literature study of which services is planned for implementation in a structure as shown in Figure 1
- Give an overview of the NWIS structure and describe the available API and communication protocols
- Selection of service(s) to evaluate.
- Plan and design experiment to collect data and review with Statens vegvesen engineers
- Perform experiment to collect data for evaluation
- Evaluate the selected service(s) on basis of the collected data from field tests
- Potentially suggest improvement of service(s) or how to adapt them to work with the interchange.

Student category: The thesis is reserved by Alexander Svindseth (IIA student employed at Statens vegvesen)

The task is suitable for online students (not present at the campus): Yes

Practical arrangements:
Statens vegvesen will give access to interchange and needed equipment for the device.

Supervision:
As a general rule, the student is entitled to 15-20 hours of supervision. This includes necessary time for the supervisor to prepare for supervision meetings (reading material to be discussed, etc).

Statens vegvesen's supervisor will be Senior Principal Engineer Ph.d Tomas Levin

Signatures:

Supervisor (date and signature):

Student (write clearly in all capitalized letters):

Student (date and signature):