SAFE COOPERATING CYBER-PHYSICAL SYSTEMS USING WIRELESS COMMUNICATION

PROTOTYPE SIMULATION TOOLS

Report type	Demonstrator
Report name	D4.3 Prototype simulation tools
Dissemination level	PU
Report status:	Final
Version number:	1.42
Date of preparation:	2018.09.28

Authors (editing)

Paul Pop, Technical University of Denmark

Fotis Foukalas, Technical University of Denmark

Revision chart and history log

Version	Date	Reason
1.0	2017-10-09	First version of document, with the structure decided
1.1	2017-10-04	ItemForms to collect prototype information decided and included
1.2	2018-08-10	Waiting links from DNV GL and KTH
1.32	2018-09-17	First complete version ready
1.41	2018-09-25	First revision round

Contents

SUMMARY AND STRUCTURE

This deliverable is a "demonstrator" deliverable from WP4, which reports on the simulation prototypes of SafeCOP. Several simulation methods and tools have been developed in SafeCOP and are used for varied tasks in the use cases, see the respective descriptions.

The deliverable is structured as follows.

- Each simulation prototype is presented via an "Item Form" which collects the main details about the prototype.
- The actual prototype is available in binary executable format and/or as source code at the repository specified in the Item Form, e.g., at https://polimicg.org/gitlab
- The Item Form contains also information (references) about the method and/or tool, and additional documentation may also be uploaded in the repository.
- We have also reported on the use of existing Open Source tools for simulation in the use cases, and then the focus is on the results of the simulation.

Finally, a detailed documentation about how to use the simulation, tool etc. is provided in particular links in the items below.

The text below presents eight items that has been developed by the following seven partners:

1. Malardalen University - MDH
2. Institute of Electronics, Computer and Telecommunication Engineering – CNR-IEIIT
3. Royal Institute of Technology Sweden – KTH
4. Politecnico di Milano - POLIMI
5. Technical University of Denmark – DTU
6. Università dell'Aquila – UNIVAQ
7. DNV GL – DNV GL

ITEM FORM MDH

1. PARTNERS INFO

1.1. PARTNERS NAME

MDH Malardalens Hogskola

1.2. CONTACT PERSONS

Full name: **Ali Balador**

Email: ali.balador@mdh.se

Mobile: +46-73-0532133

Skype: ali.balador

2. ITEM INFO

2.1. ITEM TYPE

Simulation tool.

2.2. ITEM NAME

Simulation of vehicle platooning based on open source Plexe.

2.3. REFERENCE LINKS

http://plexe.car2x.org/

https://github.com/michele-segata/plexe-sumo/tree/plexe-2.1

https://github.com/michele-segata/plexe-veins/tree/plexe-2.1

2.4. PURPOSE OF ITEM

The simulation framework combines state-of-the-art tools to tackle complex scenarios for a quantitative safety analysis. One focus are platooning scenarios, in which trucks brake coordinated and synchronized to minimize the required safety distance during mission time.

2.5. DESCRIPTION OF ITEM

PLEXE is an OMNet++ based simulation platform specially tailored for simulating platooning applications that was developed as an extension of Veins, as shown in figure 1. It provides a complete vehicular communication protocol stack based on IEEE 802.11p along with modified SUMO to accommodate realistic vehicle mobility. The vehicles in the road traffic simulator SUMO are represented by OMNet++ nodes and communicated through TraCI interface. Thus, control applications and inter-vehicle communications are handled by OMNet++ code. PLEXE already comes with ACC and some popular CACC controllers and also flexible for implementing custom made CACC algorithms. Some of its notable features are listed below:

- The developers keep it up to date with latest veins, SUMO and OMNet++ releases.
- Platoon control and vehicle injection can be done in OMNet++ initialization file. One doesn't need to deal with SUMO flows.
- Different engine models, vehicle mass, aerodynamics, kinematic parameters etc. can be tuned in the simulation.
- Vehicle tracking can be performed at PLEXE-SUMO GUI. It also provides a result analysis tool developed in R.

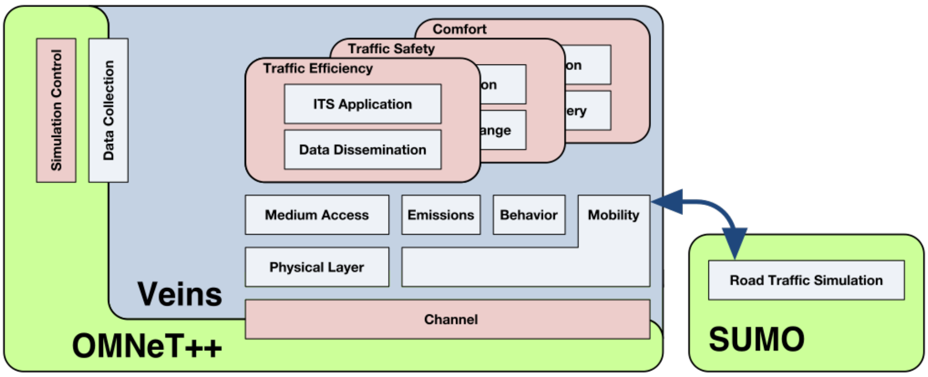


Figure 1: Veins framework

PLEXE offers necessary functionalities to simulate a wide range of scenarios to mimic real platooning environments and supports a higher level of modification both for the researchers in control theory and vehicular networking by the dint of the modularity characteristic of OMNet++ framework.

OMNeT++[1] is a C++ based, open source, and discrete event network **simulator with strong GUI** support. It can be used for modeling any system which can be described as a queuing system. **Since OMNeT++ is a general** simulation framew**ork, it does not natively** provide models for wireless network simulation. Therefore, several frameworks have been developed on OMNeT++ for sim**ulating mobile ad hoc networks.**

Each model in OMNeT++ is composed by a set of reusable components termed modules. OMNeT++ allows users to use two types of components: simple and compound modules. The functionality of each simple module is defined by C++ code, and it uses a simulation library. These simple modules can be combined like LEGO blocks to make compound modules. Simple modules **can be connected with links** which connect two gates of two different modules to each other. Gates are input and **output interfaces of modules.** Connections are created within a single level of module hierarchy. Connections spanning different hierarchy levels are not permitted, as they would prevent model reuse. Modules communicate with each other through messages, which carry arbitrary data. Messages travel through a chain of connections, starting and arriving in simple modules. Messages can contain some information

[1] "OMNeT++ home page," Available at http://www.omnetpp.org.

from other modules or be sent to the same module (self-messages) in order to **IMPLEMENT A TIMER.**

Some of the features of OMNeT++ are:

- Possibility of designing modular simulation models, which can be combined and reused flexibly.
- Composing models with any granular hierarchy.
- Availability of extensive simulation libraries that include support for input/output, statistics, data collection, random number generation and data structures.
- C++ based simulation kernel, which allows using it in larger applications.
- Graphical simulation configuration interface using NED and omnetpp.ini without requiring the use of scripts.

The main problems of this simulator in order to be a suitable simulator for VANETs are the lack of models for wireless networks, and the lack of an integrated mobility manager. Therefore, a mobility manager and a framework in order to support models for wireless communications must be installed to provide these key functionalities when attempting to achieve a realistic VANET simulation environment. In this thesis, we chose this simulator coupled with the INETMANET or MiXiM frameworks along with SUMO in order to provide realistic vehicular scenarios. INETMANET is an open-source package which provides network simulation models for OMNeT++. Although it focuses on the high level of the protocol architecture for wired and wireless communications, it also includes MAC layer protocols similar to IEEE 802.11p. Another option is to use this package in conjunction with the MiXiM package. MiXiM implemented detailed wireless NICs, so using these two packages together provides a detailed implementation at all levels of the protocol architecture. Also, SUMO is used to generate real vehicular traffic in road networks.

In summary, for each simulation using OMNeT++, the following parts must be described:

- Simple modules (provided by some packages like INETMANET, MiXiM, etc).
- Topology (using NED files).
- Simulation configuration (using omnetpp.ini).

Simulation Of Urban Mobility (SUMO)[2] is an open source, microscopic traffic simulator specially designed to handle large scale vehicular mobility. It supports three different types of elements: vehicle types, trips, and routes. A Vehicle type specifies the physical properties of a typical vehicle in the simulator. A Trip defines the departure time and the destination edge, while route expands trip by defining all the edges through which a vehicle will pass. In general, we must provide different files to SUMO in order to define the simulation map, the obstacles, and the routes used. The simulation map can be either synthetic or a real map extracted from openstreetmap.org.

[2] L. E. J. Behrisch, M. Bieker and D. Krajzewicz, "SUMO-Simulation of Urban MObility: an overview in: SIMUL 2011," in The Third International Conference on Advances in System Simulation, 2011.

2.6. TRL

Technology readiness level 3 (please avoid acronyms)

3. SAFECOP-RELATED INFO

3.1. SAFECOP EXPLOITATION AND EXTENSIONS

The PLEXE simulator is an already existing vehicular simulator previously developed in a consortium of different countries. The intention is to provide support for evaluating concepts and technologies developed by safecop.

3.2. TRL

Technology readiness level 3 (please avoid acronyms)

3.3. RELATION TO PROCESSES AND PLATFORMS

The PLEXE simulator provides a demonstrator for the runtime manger concept and safety layer developed by WP3 and 4 within safecop project.

3.3.1. Tool usage process

The tool is a C++ simulator environment. The user needs to implement the simulation using three files: NED files, C++ source files and ini files. NED files define parameters for different system component, which is defined by C++ source code. Ini file setup the scenario and parameters as well for the whole simulation.

The tool usage process is very simple, the user starts the simulator using ini file.

3.3.2. Inputs/Outputs

Inputs are entered to the simulator via ini file, which includes configuarations for the simulation.

Simulation results are saved into textual, line-oriented files. The advantage of a text-based, line-oriented format is that it is very accessible and easy to parse with a wide range of tools and languages, and still provides enough flexibility to be able to represent the data it needs to (in contrast to e.g. CSV).

3.4. USE CASES INTERACTIONS

The framework is developed within UC5.6. Scenarios simulated with the framework developed by KTH can be compared and validated.

4. BIBLIOGRAPHY

[1] A. Balador, A., Uhlemann, E., Calafate, C.T., Cano, J.-C. Supporting Beacon and Event-Driven Messages in Vehicular Platoons through Token-Based Strategies. Sensors 2018, 18, 955.

[2] A. Balador, A. Böhm, C. T. Calafate and J. Cano, "A reliable token-based MAC protocol for V2V communication in urban VANET," 2016 IEEE 27th Annual International Symposium on Personal, Indoor, and Mobile Radio Communications (PIMRC), Valencia, 2016, pp. 1-6.

[3] A. Balador, "A Token-Based MAC Protocol for Achieving High Reliability in VANET," BMW Summer School 2016 (BMWSS 2016).

[4] A. Balador, A. Böhm, E. Uhlemann, Calafate, C.T., Cano, J.-C "A Reliable and Efficient Token-Based MAC Protocol for Platooning Applications," 12th Swedish National Computer Networking Workshop (SNCNW 2016).

Item form CNR-IEIIT-1

1. Partners Info

1.1. Partners Name

CNR-IEIIT, Impara.

1.2. Contact persons

Maurizio Mongelli,

maurizio.mongelli@ieiit.cnr.it,

+39.010.6475225,

skype: mauriziomongelli

2. Item Info

2.1. Item Type

Simulation tool.

2.2. Item Name

Simulation of vehicle platooning based on open source Plexe.

2.3. Reference Links

https://github.com/mopamopa/Platooning point 2; please refer to the Readme.md

http://plexe.car2x.org/ [Mon18]

2.4. Purpose of Item

Simulate platooning scenarios under different constraints and system parameter settings.

2.5. Description of Item

Plexe[3] is an extension of the popular Veins vehicular network simulator which permits the realistic simulation of platooning (i.e., automated car-following) systems. It features realistic vehicle dynamics and several cruise control models, permitting the analysis of control systems, large-scale and mixed scenario, as well as networking protocols and cooperative maneuvers. It is free to download and easy to extend [San17].

An automated scheme has been provided in order to set Plexe runs according to different parameters and register results accordingly (e.g., collision under specific circumstances). The registered log is then used to feed data into a data analytics process.

[3] http://plexe.car2x.org/.

2.6. TRL

TRL 4.

3. SAFECOP-RELATED INFO

3.1. SAFECOP EXPLOITATION AND EXTENSIONS

Structure

The basic structure of Plexe defines the simulation of the platoon dynamics under specific conditions, such as initial speed, distance, Quality of Service of the communication channel [San17]. In order to collect information about performance under different conditions, the Plexe executable is inserted into a main loop (Bash script in Linux), in which parameters are changed according to predefined ranges, such as, number of vehicles in [3, 8], Packet Error Rate in [0.2, 0.8]. The inherent results of each run in Plexe are registed and post-processed at the end of the main loop. This involves the automatic setting of parameters in several configuration files of the plexe environment. The attached sw shows all the scripts involved in the automation of Plexe.

Use in SafeCOP

The automation of the simulator has been developed from scratch. The data generated from the simulations is used in WP4 to drive knowledge extraction of vehicle platooning conditions: D4.1_VAL explains the methodology and D4.2 an example applicable to the run time manager.

Example

[Mon18] introduces the use of machine learning with rule generation to validate collision avoidance in vehicle platooning. Cooperative Adaptive Cruise Control is under test over a range of system parameters including speed and distance of the vehicles as well as packet error rate of the communication channel. Safety regions are evidenced on test data with statistical error very close to zero. Performance evaluation is based on the automation of the plexe environment outlined here.

3.2. TRL

Technology validated in lab: TRL 4.

3.3. RELATION TO PROCESSES AND PLATFORMS

The generated sample paths of the system drive performance forecast through data analytics. The inherent prediction model may be used as a unit of the run time manager to predict unsafe behaviours at run time.

3.3.1. Tool usage process

Collection of samples acquired from the simulator are fed into a data analytics platform (Rulex). The platform is customized so as to derive an intelligible prediction model of safety/performance metrics. The model may be then “interrogated” within the simulator for further testing and data generation.

3.3.2. Inputs/Outputs

Input: platooning system parameters (e.g., number of vehicles, communication delay, etc.).

Output: sample paths of the system over time.

3.4. Use Cases Interactions

UC3: collision/instability prediction of vehicle platoon.

UC5: congestion/pollution prediction of vehicle traffic scenarios.

4. Bibliography

[San17] S. Santini, A. Salvi, A. S. Valente, A. Pescapé, M. Segata and R. Lo Cigno, "A Consensus-Based Approach for Platooning with Intervehicular Communications and Its Validation in Realistic Scenarios," in IEEE Transactions on Vehicular Technology, vol. 66, no. 3, pp. 1985-1999, March 2017. doi: 10.1109/TVT.2016.2585018..

[Mon18] E. Ferrari. A. Fermi, M. Mongelli, M. Muselli, “Identification of Safety Regions in Vehicle Platooning via Machine Learning,” 14th IEEE Internat. Work. on Fact. Commun. Sys. WFCS 2018.

ITEM FORM CNR-IEIIT-2

1. PARTNERS INFO

1.1. PARTNERS NAME

CNR-IEIIT, Impara.

1.2. CONTACT PERSONS

Maurizio Mongelli,

maurizio.mongelli@ieiit.cnr.it,

+39.010.6475225,

skype: mauriziomongelli

2. ITEM INFO

2.1. ITEM TYPE

Simulation tool.

2.2. ITEM NAME

Discrete time simulation of vehicle platooning through differential equations.

2.3. REFERENCE LINKS

https://github.com/mopamopa/Platooning point 1; please refer to the Readme.md

Prediction collision in platooning:

https://www.dropbox.com/s/j46du6hnbuwbavl/Machine%20learning-platooning%20example.pdf?dl=0.

Prediction of string stability in platooning:

https://www.dropbox.com/s/fny0pb7db407uc9/D3.4_CNR-IEIIT%26Impara.pdf?dl=0.

2.4. PURPOSE OF ITEM

Simulate platooning scenarios under different constraints and system parameter settings.

2.5. DESCRIPTION OF ITEM

Discrete time version of differential equations of [Xu14] are used to generate sample paths of a platooning system. Asynchronous events (e.g., the transmission of an 802.11p packet) may be also included within the main loop as well as the adoption of QoS paramaters along the communication channel, such as transmission delays between cars.

2.6. TRL

TRL 4.

3. SAFECOP-RELATED INFO

3.1. SAFECOP EXPLOITATION AND EXTENSIONS

Exploitation: none.

Extensions and development: the software has been built from scratch.

Intended use: generate sample paths of the system to drive data analytics.

3.2. TRL

Technology validated in lab: TRL 4.

3.3. RELATION TO PROCESSES AND PLATFORMS

The generated sample paths of the system drive performance forecast through data analytics. The inherent prediction model may be used as a unit of the run time manager to predict unsafe behaviours at run time.

3.3.1. Tool usage process

Collection of samples acquired from the simulator are fed into a data analytics platform (Rulex). The platform is customized so as to derive an intelligible prediction model of safety/performance metrics. The model may be then “interrogated” within the simulator for further testing and data generation.

3.3.2. Inputs/Outputs

Input: platooning system parameters (e.g., number of vehicles, communication delay, etc.).

Output: sample paths of the system over time.

3.4. USE CASES INTERACTIONS

UC3: collision/instability prediction of vehicle platoon.

UC5: congestion/pollution prediction of vehicle traffic scenarios.

4. BIBLIOGRAPHY

[Xu14] L. Xu, L. Y. Wang, G. Yin and H. Zhang, "Communication Information Structures and Contents for Enhanced Safety of Highway Vehicle Platoons," in IEEE Transactions on Vehicular Technology, vol. 63, no. 9, pp. 4206-4220, Nov. 2014.

ITEM FORM KTH

1. PARTNERS INFO

1.1. PARTNERS NAME

KTH Royal Institute of Technology

1.2. CONTACT PERSONS

Professor Karl Meinke,

School of Computer Science and Communications

karlm@kth.se

mobile: 076 223 8679

skype: karl_meinke

2. ITEM INFO

2.1. ITEM TYPE

Other

2.2. ITEM NAME

Simple platooning simulator

2.3. REFERENCE LINKS

https://gits-15.sys.kth.se/emarten/PlatoonSim

2.4. PURPOSE OF ITEM

The main purpose of this item is to provide a point-mass model simulator for vehicle platoon dynamics that can be used as a demonstrator for safety analysis of distributed CO-CPS algorithms. The main type of analysis that we aim to support is *systems-of-systems testing* for CO-CPS based on *formal temporal logic requirements*. The actual analysis will be carried out within the D4.4 item Quantitative Safety Analysis (QSA) for CO-CPS.

2.5. DESCRIPTION OF ITEM

The final item is a Java implementation of a lightweight vehicle simulator that be configured with typical parameters of a truck (weight, length, width, braking and engine capacity). This application is termed **KTH-PlatoonSim**. The simulator is object-oriented, so that several truck objects can be combined into a platoon object. Besides simulating basic vehicle dynamics, the simulator also supports a model of V2V communication using single hop and multi-hop TDMA communication between individual vehicles. Aspects of the IEEE 802.11p protocol are included, though the simulation is not as detailed as e.g. Veins. In addition, the V2V communication model supports an ambient communication noise model by randomized packet loss. This model is based upon empirical road measurements carried out by SafeCOP partner Qamcom within the previous Relcommh project.

A design criterion of this item was the need for a lightweight high performance simulator, so that very many simulation runs could be executed in a given time frame, e.g. using an HPC platform. Thus our execution rate is approximately 100 times real life (1 minute of simulation = 100 minutes driving time). This has been achieved for platoons in the size range 2-8 vehicles. This compares with a heavyweight simulator such as commercial tool TruckMaker, which has about 4-10 times execution speed-up, depending on the model. Another design criterion was a simple uniform block structure that allows us to plug-and-play with different control algorithms. Finally, we have complete internal models of environment physics, such as road surface friction, wind resistance, message packet loss rates, etc. The alternative would have been to use an external simulation engine, such as the Unity game engine. This approach was taken to increase the transparency and maintainability of the simulator.

To satisfy these requirements, several simplifications have been made, relative to a heavyweight simulator. The main ones are the lack of: engine, powertrain and suspension models. Trucks cannot currently be articulated. There is no steering model either, so that only 1-dimensional motion is modeled.

The architecture of a single vehicle in KTH-PlatoonSim is show in Figure 2: Single Vehicle Model. This diagram shows the brake-by-wire (BBW) sub-architecture consisting of a brake-torque controller, a global brake controller, ABS and vehicle wheel models (for a 4 wheel vehicle). External to this are a co-operative adaptive cruise control (CACC) algorithm, as well as odometry and wireless communication components.

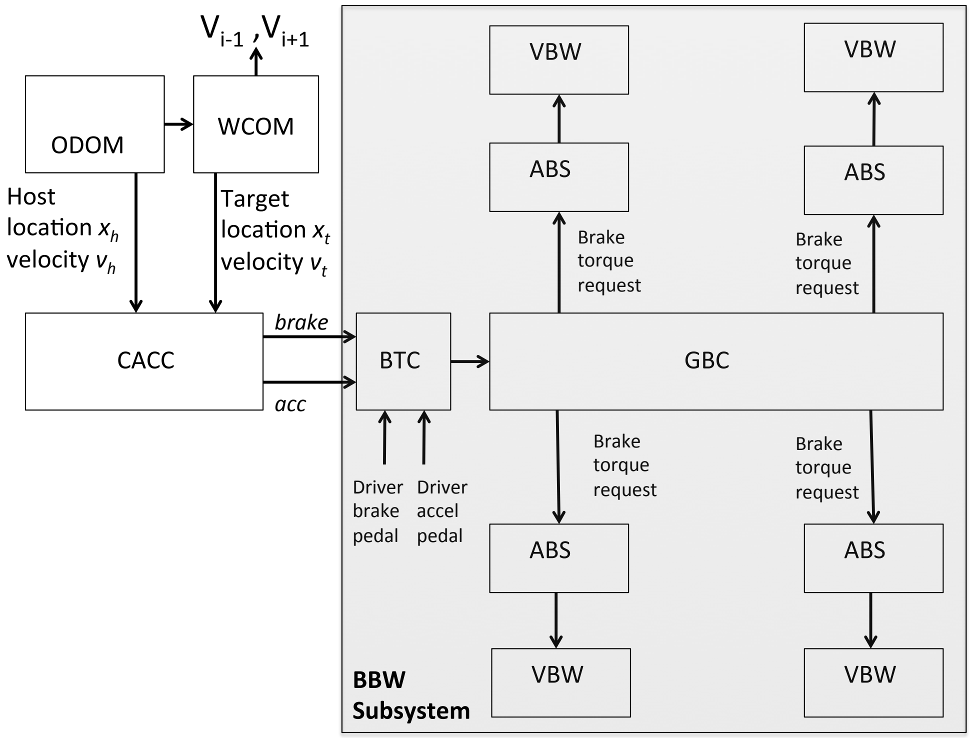


Figure 2: Single Vehicle Model

In Figure 3: Wireless Communication Architecture we show the architecture of the wireless communication system between vehicles. This communication framework assumes wireless broadcast and point-to-point multi-hop communication between the vehicles in the platoon. A slotted TDMA scheme based on ideas from [2] is implemented: to avoid communication collisions, each vehicle V_i is allowed to transmit only in its own TDMA slot.

As communication is broadcast-based, receiving vehicles can lose packets independently during a broadcast operation. Thus a packet can be received by one vehicle and lost by another. In a platoon of N vehicles, for any sender V_i and receiver V_j (where $0 \le i,j \le N-1, i \ne j$) let $d=|i-j|$ correspond to the distance between the sender and receiver. The probability P in percent of a message being lost is P (message lost) = P ERbase+increase·(d−1). Note that with the values from the Relcommh road test, the probability of message loss (from the LV to the last vehicle) is 100% in a platoon of eight vehicles or more; hence every message is lost (unless e.g. multi-hop communication is used).

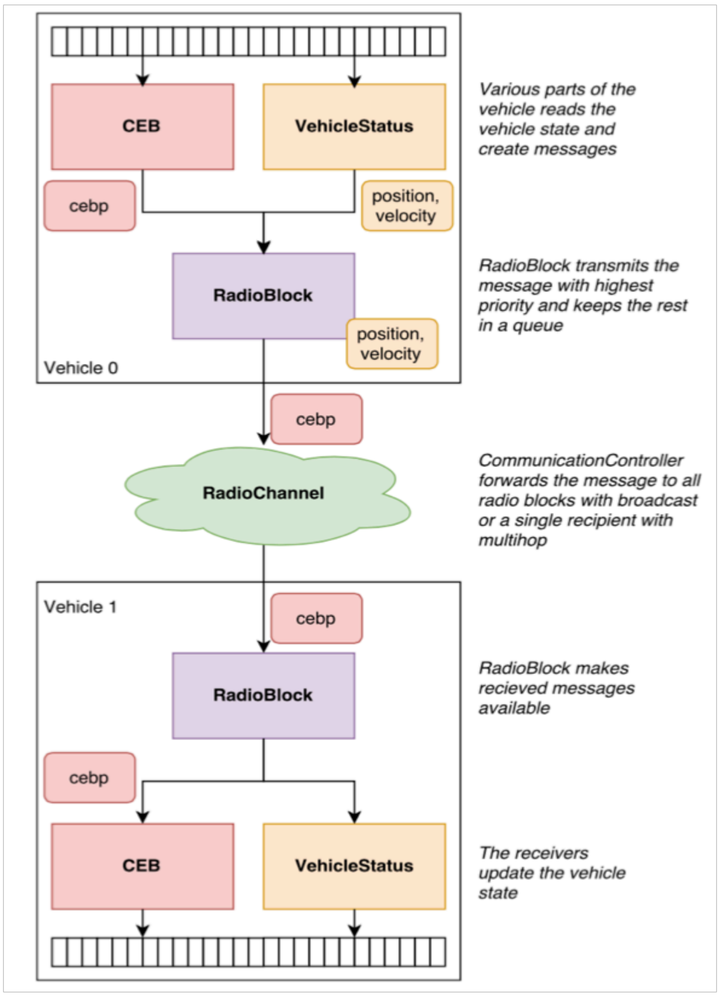


Figure 3: Wireless Communication Architecture

In Figure 4: a 2-vehicle platoon showing critical design parameters, we show a typical configuration of vehicle models together with safety parameters such as distance headway x_r. Also time headway can be used to measure inter-vehicle gaps.

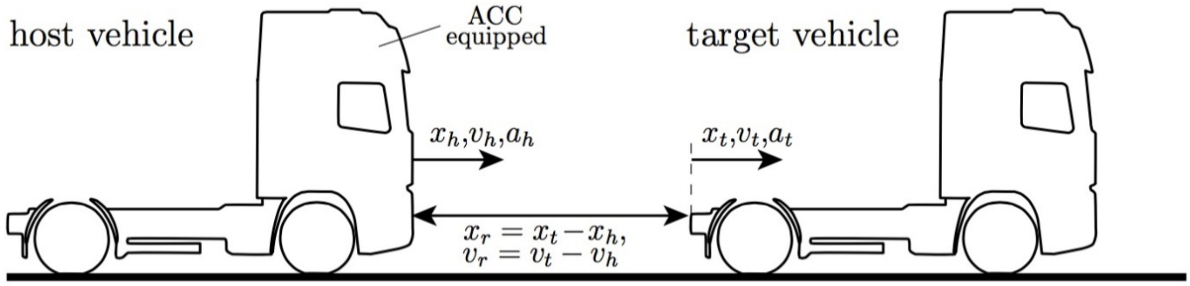


Figure 4: a 2-vehicle platoon showing critical design parameters

In Figure 5: A visualisation of KTH-PlatoonSim in KTH-PlatoonView based on Unity game engine, we show a visualization of the result of executing KTH-PlatoonSim using the Unity game engine (see https://unity3d.com/). Unity is a cross platform game engine suitable for developing 2D and 3D games, with a large community online. The visualization tool is called KTH-PlatoonView. This additional tool was developed to demonstrate the results of testing KTH-PlatoonSim with LBTest. It is especially useful to understand how failed test cases (identified by LBTest) have evolved over time. KTH-PlatoonView gives a 3D visualisation that accurately reproduces each KTH-PlatoonSim execution. It does this by switching off the internal Unity physics so that there is no contradiction between the KTH-PlatoonSim physics and the Unity physics.

Figure 5: A visualisation of KTH-PlatoonSim in KTH-PlatoonView based on Unity game engine.

2.6. TRL

TRL 3 (EU definition)

3. SAFECOP-RELATED INFO

3.1. SAFECOP EXPLOITATION AND EXTENSIONS

The platooning simulator is an extension of an already existing brake-by-wire simulator developed previously at KTH. The intended use is to provide support for CO-CPS simulation (specifically platooning) as part of the safety certification process for CO-CPS.

3.2. TRL

TRL 3 (EU definition)

3.3. RELATION TO PROCESSES AND PLATFORMS

The platooning simulator provides a demonstrator for the QSA safety assurance techniques developed as SafeCOP processes as a D4_4 item.

3.3.1. Tool usage process

The tool is a standalone Java application. It is driven by a sequence of accelerator, brake and emergency brake commands. There are currently two output APIs, both of which are text file formats. One output file format supports the extraction of driving behaviours that is suitable for the KTH testing tool LBTest. The other output file format produces a text file with frame information that can be used to drive visualization in the open-source game platform Unity.

The tool usage process is very simple, the user starts the simulator and types in a command sequence as a comma separated list of commands. The simulator executes these commands and produces in the chosen output file format, the dynamics of the platoon simulation (speeds, acceleration values, distances, crash incidences if any).

3.3.2. Inputs/Outputs

Input: lead vehicle driver behaviours i.e. gas, brake and steering commands

Input: environmental conditions, e.g. ambient packet loss

Output: vehicle dynamics for each platoon vehicle, displacement, velocity, acceleration, etc.

3.4. USE CASES INTERACTIONS

The QSA methodology will be exemplified in UC5.6 using the platooning simulator to show that this is a generic methodology that can be applied to the other CO-CPS use cases. Different safety requirements will be formulated for different uses cases on a platooning simulator (e.g. emergency braking). Different system parameters will be identified (e.g. time headway, distance headway), and a QSA will be conducted to exemplify the approach.

4. BIBLIOGRAPHY

[1]. Karl Meinke: *Learning-Based Testing of Cyber-Physical Systems-of-Systems: A Platooning Study*. EPEW 2017: 135-151

https://link.springer.com/chapter/10.1007/978-3-319-66583-2_9

[2]. Bohm, A., Jonsson, M., Kunert, K., Vinel, A.: *Context-Aware Retransmission Scheme for Increased Reliability in Platooning Applications*, pp. 30–42. Springer International Publishing, Cham (2014), http://dx.doi.org/10.1007/ 978-3-319-06644-8_4

[3]. Carl Bergenhem, Fabian Ström, Karl Meinke, *Quantitative Safety Analysis of a Coordinated Emergency Brake Protocol for Vehicle Platoons*, to appear in Proceedings ISoLA'18, 8th International Symposium On Leveraging Applications of Formal Methods, Verification and Validation, Springer LNCS, 2018.

[4] Karl Meinke, Muddassar A. Sindhu: Incremental Learning-Based Testing for Reactive Systems. TAP 2011: 134-151. https://link.springer.com/chapter/10.1007/978-3-642-21768-5_11

ITEM FORM POLIMI

1. PARTNERS INFO

1.1. PARTNERS NAME

POLIMI, CNR

1.2. CONTACT PERSONS

Pietro Fezzardi,

pietro.fezzardi@polimi.it,

skype: pietro_fezzardi

2. ITEM INFO

2.1. ITEM TYPE

Tool

2.2. ITEM NAME

Veins Extension to support Lane Detectors

2.3. REFERENCE LINKS

https://polimicg.org/gitlab/fez/safecop

https://polimicg.org/gitlab/fez/veins

https://polimicg.org/gitlab/fez/omnetpp

2.4. PURPOSE OF ITEM

Extensions to the Veins framework to be used with SUMO and Omnet++ to simulate components of the UC5 V2X scenario.

2.5. DESCRIPTION OF ITEM

An extension of the Veins traffic simulation framework, based on SUMO and Omnet++, to support the use of cameras to provide inputs to the decision tools developed in UC5.

2.6. TRL

TRL 4

3. SAFECOP-RELATED INFO

3.1. SAFECOP EXPLOITATION AND EXTENSIONS

SUMO, Omnet++, and Veins are well-known simulation tools used by many projects. The extension of Veins, has been designed, developed, and implemented ad hoc for SafeCOP. It is used to provide a model of the UC5 scenario that can be simulated at larger scale (i.e., with more than one traffic light and with real data from roadside cameras).

3.2. TRL

TRL 4

3.3. RELATION TO PROCESSES AND PLATFORMS

The simulator components will be used to provide input to test the SafeCOP reference platform employed in UC5 to provide safety to the traffic light – sensors – vehicles system of systems.

3.3.1. Tool usage process

The tool is used in combination with Veins. In UC5, we will provide prebuilt scenarios. In general, it is possible to use the provided camera component to monitor any road lane in a SUMO simulation from Omnet++, and transmit the monitored information (e.g., number of cars per time unit) through Omnet++ implemented networks.

3.3.2. Inputs/Outputs

The component provides the number of cars per time unit in the simulated enviroment passing along a road lane.

3.4. USE CASES INTERACTIONS

The component is used to provide a model of the UC5 scenario that can be simulated at larger scale (i.e., with more than one traffic light and with real data from roadside cameras).

4. BIBLIOGRAPHY

[1] Christoph Sommer, Veins, http://veins.car2x.org

[2] Filippo Leveni, Lane Detector contribution to Veins, https://github.com/ineveLoppiliF

ITEM FORM DTU

1. PARTNERS INFO

1.1. PARTNERS NAME

Technical University of Denmark (DTU)

1.2. CONTACT PERSONS

Paul Pop, email: *paupo@dtu.dk*

Fotios Foukalas, email: *fotisf@dtu.dk*

2. ITEM INFO

2.1. ITEM TYPE

Simulator in Matlab

2.2. ITEM NAME

SL D2D V2V Communications Simulator

2.3. REFERENCE LINKS

A link to the technical report and the tool is the following:
https://github.com/NestorBonjorn/3GPP_SPS_enhanced

2.4. PURPOSE OF ITEM

The simulator is used to simulate both link level communication among two different vehicular user equipment (vUE) and system level of multiple vUEs. This is helpful to provide a cooperative framework for ultra-reliability and low latency wireless communications.

2.5. DESCRIPTION OF ITEM

This item is a simutation of a Sidelink (SL) Device-to-Device (D2D) communications that is considered an extension towards enhanced 5G vehicular communications. In particular, the extension provides two different modes of resource allocation for in-coverage and out-of-coverage use cases. The simulation item is actually a link level simulator designed according to physical layer specification too. The particular work is considered as Rel.14 of 3GPP. Further to this direction, scheduling is also developed for the modes 3 and 4 resource allocation mechanism. To be specific, a semi-persistent scheduling (SPS) mechanism is incorporated into the system level simulator to measure the collision probability for a number of vehicular user equipment. The particular scheculting mechanism is considered a baseline design to provide a cooperative framework, where the vehicles could retain a specific reliability and latency in both in coverage and out of coverage scenarios. Such a simulation is useful for all type of V2X use cases addressed by Safecop.

2.6. TRL

TRL3

3. SAFECOP-RELATED INFO

3.1. SAFECOP EXPLOITATION AND EXTENSIONS

The simulator is considered related to the following standardization activities towards 5G V2X communications:

- 3GPP, "Technical Specification Group Services and System Aspects; Study on enhancement of 3GPP Support for 5G V2X Services (Release 15)", TR 22.886 v15.1.0, Mar. 2017
- 3GPP R1-162363, "On Sensing Design Details for Sidelink V2V Communication", Intel Corporation, 3GPP TSG RAN WG1 meeting #84bis, Korea, Apr. 2016.

As mentioned before, using the particular simulator, which just released from the 3GPP body, not available somewhere though, we can build cooperative techniques for efficient wireless communications in vehicular type of CPS.

3.2. TRL

TRL4

3.3. RELATION TO PROCESSES AND PLATFORMS

The simulator is useful to incorporate V2V interface among vehicles to all V2X use cases simulating the message payload, latency and reliability. The air interface is considered towards 5G. Safecop platform can incorporate this functionality into the overall system.

3.3.1. Tool usage process

This document describes the simulator presented in the following Github repo: https://github.com/NestorBonjorn/3GPP_SPS_enhanced. This simulator calculates transmission collision probabilities between UEs using the semi-persistent scheduling (SPS) procedure used for sidelink communications in mode 4, as standardized by 3GPP in Release 14. For that, we simulate a set of UEs that transmit to each other in a periodic manner. Moreover, a proposed enhancement of the current standardized system is implemented and can also be simulated. The standardized version is simulated if the `proposal` flag is set to `false` and our proposed enhancement is simulated if the `proposal` flag is set to `true`.

3.3.2. Inputs/Outputs

The simulator starts by defining a set of simulation-related parameters, such as the flag (`proposal`) indicating whether we simulate our proposed enhancement or not, the simulated number of vehicles (`num_vehicles`), the number of simulations (`num_simulations`) from which we will take the average in order to obtain the final result, and the number of subframes simulated in each simulation (`num_subframes_simulated`). We also initialize the variable that will store the different simulation results as a matrix of zeros (`cumul_cat`).

```
% Flag indicating whether we use our proposed approach or not
proposal = true;
% Number of simulated vehicles
num_vehicles = 10;
% Number of simulations
num_simulations = 10;
% Number of simulated subframes per simulation
num_subframes_simulated = 1000000;
% It will store the different simulation results
cumul_cat = zeros(num_subframes_simulated/1000,num_simulations);
```

3.4. Use Cases Interactions

The tool will be used in UC3 to 5 that are relevant to vehicular use cases. The 3GPP based simulator can be used to all those use cases to simulate the performance of V2V and V2I links.

4. Bibliography

Additional reference is as follows:

- Chen, Shanzhi, et al. "Vehicle-to-Everything (v2x) Services Supported by LTE-Based Systems and 5G." IEEE Communications Standards Magazine 1.2 (2017): 70-76.
- 3GPP, “Technical Specification Group Services and System Aspects; Study on enhancement of 3GPP Support for 5G V2X Services (Release 15)”, TR 22.886 v15.1.0, Mar. 2017

Item form UNIVAQ

1. Partners Info

1.1. Partners Name

University of L'Aquila (UNIVAQ).

1.2. Contact persons

Elena Cinque, PhD
Department of Information Engineering, Computer Science and Mathematics
elena.cinque@univaq.it

Francesco Valentini, PhD
Department of Information Engineering, Computer Science and Mathematics
francesco.valentini1@univaq.it

Professor Marco Pratesi,
Department of Information Engineering, Computer Science and Mathematics
marco.pratesi@univaq.it
phone: (+39)-0862.434624
mobile: (+39)-338.6298932

2. Item Info

2.1. Item Type

A mechanism (and evaluation with a simulation tool).

2.2. Item Name

Evaluation of IEEE 802.11p channel congestion and its mitigation for Co-CPS safety.

2.3. Reference Links

This list to be updated with new publications during the project:

E. Cinque, F. Valentini, A. Iovine and M. Pratesi, "An adaptive strategy to mitigate instability in the ETSI DCC: Experimental validation", *http://ieeexplore.ieee.org/document/7972223/*

http://veins.car2x.org

https://github.com/elecin/D43_ItemForm_UNIVAQ_TLC

2.4. Purpose of Item

The main purpose of the item is the evaluation of the radio channel congestion for vehicular communications based on the IEEE 802.11p standard, followed by an estimation of the impact on safety services and maybe the proposal of a congestion mitigation strategy.

2.5. Description of Item

The IEEE 802.11p protocol has been specifically customized for vehicular cooperation supporting direct communication, also in absence of network coverage, with low latency and traffic prioritisation. A frequency band around 5.9GHz has been reserved to this kind of communications divided into 10MHz channels with different organization in the European and US standards.
The mainly used protocol stacks that relies on the 802.11p are the ETSI ITS-G5 in Europe and the WAVE in the US and, both of them, use a periodic transmission of one-hop broadcast messages to exchange information between the road users and the roadside infrastructure.
When the number of vehicles grows higher there will be a heavy impact on the channel load and the functionality of both safety and traffic-management C-ITS applications can be compromised without proper congestion control mechanism.
Some mechanisms or proposals of congestion mitigation already exist and should be evaluated; moreover a collaborative solution to the congestion issue should be investigated.
This item is linked to the TB030 technical brick.

2.6. TRL

TRL 2 (technology concept formulated).

3. SAFECOP-RELATED INFO

3.1. SAFECOP EXPLOITATION AND EXTENSIONS

We focused on the US WAVE and the ETSI ITS-G5 stacks [1] (both based on IEEE 802.11p), as they are the most consolidated options nowadays, whereas usage of the forthcoming 5G networks is still in the experimentation phase.
After a preliminary assessment through real world COTS devices, we evaluated the channel usage in a simulation environment based on OMNeT++ and a widely used framework for the WAVE stack, i.e. Veins, for which an extension (not deeply tested) for the ETSI ITS-G5 stack is available, too. We are considering dense scenarios, to assess the impact of the channel load on the safety of CPSs. Some mechanisms for congestion mitigation are described in the literature: cooperative strategies based on them can be proposed, coherently with the SafeCOP approach.

3.2. TRL

TRL 3 (experimental proof of concept) or TRL 4 (EU definition).

3.3. RELATION TO PROCESSES AND PLATFORMS

As also reported in the Deliverable 3.1 (802.11p limitations and drawback paragraph) the data congestion should affect safety message broadcast repetition time reducing it below acceptable values. We want to point out that also an excessive reduction of the message rate could compromise the Safety of the CPSs since some applications are going to use them to collect and exchange information. The Runtime Manager could tune the broadcast reduction level, eventually in a cooperative way with other CPSs, to avoid a degradation of the current Safety Level.

3.3.1. Tool usage process

The tool is based on Veins simulator, where the SUMO mobility simulator is used in combination with Omnet++ network simulator.

First, the road scenario has been implemented in SUMO. Specifically, the traffic scenario generation can be initiated:

1. *manually, writing the xml network file;*
2. *through NETEDIT editor, which provide a GUI to graphically draw a road network scenario with junctions and links;*
3. *using real world maps from OpenStreetMaps and by converting them into the SUMO format.*

In our preliminary test the first method is used in combination with the second one.

Thus, we customized the network simulator by adding to the "MAC-layer" module the code to collect the Channel Busy Ratio (CBR) statistics (referring to the VeINS/OMNet++ project repository, the path to the modules is veins-veins-x/src/veins/modules/mac/ieee80211p). Also, an object of type cOutVector is defined to save CBR time series data (referred to as output vectors) to a output vector file.

3.3.2. Inputs/Outputs

Input: vehicular scenario settings, e.g. roads and buildings topology, number of vehicle flows, speed limit for each road.

Input: vehicle flow dynamics and settings, e.g. number of vehicle for each flow, departure point, arrival point, input frequency of new vehicles.

Input: vehicle dynamics and settings, e.g. vehicle length, average speed, max acceleration.

Input: Base Safety Messages (BSMs) features, i.e. length and transmission rate.

Outputs: Channel Busy Ratio (CBR) statistics; Packet Delivery Ratio (PDR) and transmitted messages statistics.

3.4. Use Cases Interactions

This item form could be applied to every use-case that consider the 802.11p as the communication standard between the CPSs.

4. Bibliography

This list to be updated with new publications during the project:

8. E. Cinque, F. Valentini, A. Iovine and M. Pratesi, "An adaptive strategy to mitigate instability in the ETSI DCC: Experimental validation", 2017 15th International Conference on ITS Telecommunications (ITST), Warsaw, 2017, pp. 1-6.

9. S. Chiocchio, A. Persia, F. Valentini, E. Cinque, M. Pratesi, F. Santucci, "Integrated Simulation Environments for Vehicular Communications in Cooperative Road Transportation Systems", in Proc. of the 2nd URSI AT-RASC, Gran Canaria, June 2018.

10. E. Cinque, F. Valentini, M. Pratesi, S. Chiocchio, A. Persia, "Analysis and experimental characterization of channel congestion control in vehicular networks", in Proc. of the International Symposium on Networks, Computers and Communications (ISNCC), Rome, June 2018.

ITEM FORM DNV GL

1. PARTNERS INFO

1.1 PARTNERS NAME

DNV GL

1.2 CONTACT PERSONS

Jon-Arne Glomsrud; e-mail: *jon.arne.glomsrud@dnvgl.com*

Ovidiu Valentin Drugan; e-mail: *ovidiu.drugan@dnvgl.com*

2. ITEM INFO

2.1 ITEM TYPE

Tool

2.2 ITEM NAME

Scenario Manager (SM)

2.3 REFERENCE LINKS

https://gtr-digitalassurance.visualstudio.com/SafeCOP_ScenarioManager/_git/SafeCOP_ScenarioManager

2.4 PURPOSE OF ITEM

To achieve automatic and efficient simulation based testing, a ***Scenario Manager*** *(SM) is needed to generate test scenarios and automatically evaluate the scenario outcome.*

2.5DESCRIPTION OF ITEM

The SM will connect to a Digital Twin (DT) adapted to the use case, that also includes the test target SW. The SM Automatic Generator (SM-AG), will generate a scenario by setting a number of model parameters in the DT and start and eventually stop stepping the scenario simulation. The SM Automatic Evaluator (SM-AE) will automatically monitor the scenario, evaluate the goodness of the outcome and report/log the scenario score. The SM-AG is pre-configured to traverse through a large set of scenarios and each scenario will be paired with its SM-AE score.

A scenario simulation can be developed and used for testing platooning behaviour and its dependency on the wireless communication in the maritime use case.

The use case is a Manned Surface Vessel (MSV) and one or several Unmanned Surface Vessels (USV), platooning to do bathymetric measurements. The MSV has wireless communication with the USV where critical information like path to follow, max speed, configuration parameters, manual remote operation commands, MSV position and other information can pass. The USV is following the commanded path at the maximum speed, and if the distance to the MSV is too short, the USV will halt until the distance is safe to proceed its mission. If the remote operator needs, the USV can be controlled fully remotely. If the communication is down, the USV will act autonomously and go to a defined safe state.

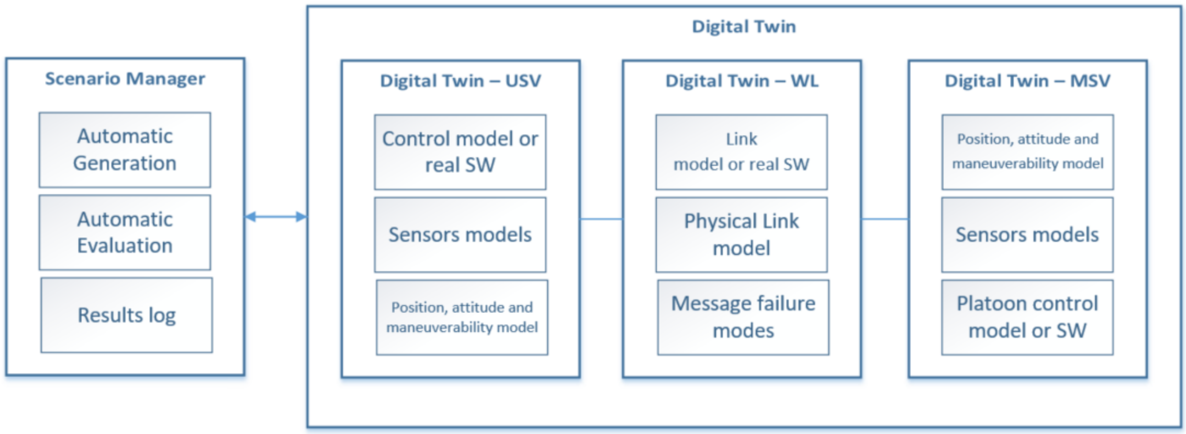
A Digital Twin (DT) (model) of an Unmanned Surface Vessel (USV) and of a Manned Surface Vessel (MSV) can be established, to simulate the typical behaviour. The DT of the USV/MSV can to a simple degree emulate the control systems or the captain manoeuvres, and certain control parameters can be varied to test out different control strategies in the platooning. As an extension, parts of the real control systems could be included instead of the emulated functions.

The Wireless Link (WL) is a critical component in the platoon control that needs to meet requirements on reliability, timeliness, security, etc. A WL-DT can be developed to model the WL and possibly connect to the real link SW. The WL-DT can simulate typical failure modes to test the effect on the platooning performance.

To efficiently run the complete DT and simulations, a Scenario Manager (SM) with an Automatic Generation (AG) of scenarios can be used. The SM-AG will generate scenarios varying platooning scenarios, control parameters and WL failure modes.

To facilitate automatic testing, also an Automatic Evaluation (AE) needs to be developed into the SM. The SM-AE needs to produce a score on the acceptable platooning performance to rate the outcome of the simulation.

The main idea is to use the complete simulation to test and verify different scenarios in a platooning operation, with different control parameters and possible failure modes of the WL. It is then possible to develop safe platooning strategies, safe control parameter values and quantitative WL requirements, which in turn can be used when testing the WL target SW.



2.6TRL

Technology validated in lab: TRL 4.

3. SAFECOP-RELATED INFO

3.1SAFECOP EXPLOITATION AND EXTENSIONS

A prototype/concept of an SM for automatic simulation based testing will be exploited and used in SAFECOP for the boats in the cooperative bathymetry UC. The specific range of scenarios to simulate, i.e. the pre-configured input to the SM, will be developed for the specific test target. Also a version of the SM-AE will be developed for the specific test target. These are both extensions or specific developments of the concept. The inteded use of the configured SM is to efficiently test and demonstrate critical system parts in the use case (in this case the boats in the cooperative bathymetry UC).

3.1.1 Scenario Manager

The SM consists of an AG, AE and Results log module described below

Automatic Generation of scenarios

The AG can, based on pre-set scenario space parameter ranges, automatically generate scenarios. The intention is to make the test process efficient. Since the scenario parameters might be many and the possible number of combinations of parameters giving unique scenarios, might be very large, the AG need to reduce the needed scenarios to test to reach a conclusion to a minimum. A conclusion is firstly that a scenario test fails ASAP (given the scenario is within the capable operation area of the control systems) and secondly that the tested scenarios together bring confidence in that the control systems are capable of always handling the operation safely. This is achieved by firstly that there are no failed scenario tests and secondly that the tested scenarios together are representative and complete for defining the planned operative domain or planned failure tolerance in the system.

To solve this, the AG needs to:

1. search efficiently for failed scenario test performance. This can be some sort of sensitivity search, i.e. the next scenario is chosen based on previous scenario results.
2. Measure or show how the scenario space has been covered so far.

Both these needed abilities are important research tasks to be solved to efficiently and completely test autonomous functions.

Automatic Evaluation of scenarios

The AE will be a score function including the following elements:

1. The USV deviation from the desired path, both lateral distance and heading vs path course
    a. A path must be followed to control the platooning and avoiding shallow areas or even land, or coming too close to any object.
    b. A payload can be a bathymetric sonar with an athwartships beam orientation. A lateral deviation will reduce the coverage of the desired footprint. A heading vs path course deviation might reduce the width of the covered footprint. This motivates including this score
2. The deviation from the max bathymetric speed. The safety aspects are handled by possible deviation from the path
3. The USV distance from the MSV
    a. For platooning safety, the MSV and USV must always keep a minimum distance apart and this must obviously be included.

Results logging

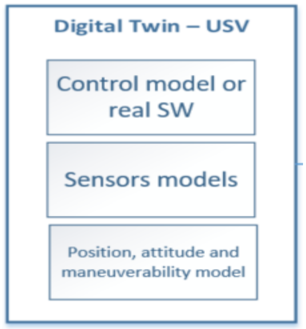
All results are logged by storing the scenario parameter vector with the score, including the individual score variables. Also, the digital twin states can be stored if needed, for investigation.

3.1.2 Digital Twin

Digital Twin of USV

The DT-USV consists of

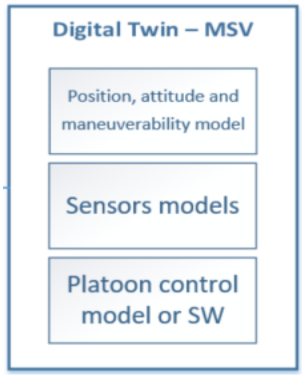
1. A model for USV control and its most important parameters and behaviors. Could include real SW if this is desired to be tested.
2. models of critical sensors and possible failure modes
3. models of the position/speed, attitude (heading, course, roll/pitch/draft (if needed))



Digital Twin of MSV

The DT-MSV consists of:

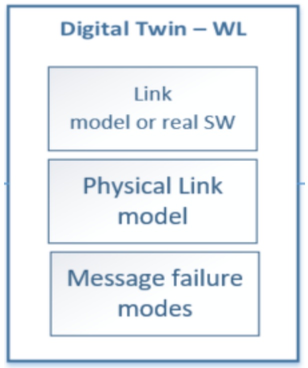
1. models of the position/speed, attitude (heading, course, roll/pitch/draft (if needed))
2. models of critical sensors and possible failure modes
3. a model for platoon control and its most important parameters and behaviors. Could include real SW if this is desired to be tested.



Digital Twin of Wireless Link

The DT-WL consists of:

1. a model of the link functionality. Could include parts of the real SW if this is desired to test
2. a model of the physical link (wireless transmission over the air and through antennas, etc=
3. a model of the message failure modes (alteration, lost, fake, delays, changed sequence, etc)



3.2 TRL

Technology validated in lab: TRL 4.

3.3 RELATION TO PROCESSES AND PLATFORMS

The generated sample simulations of the boats during a cooperative bathymetry operation. The simulations will be used to testing the behaviour of the run time manager for evaluating unsafe behaviours at run time.

3.4 Tool usage process

The current version of the file requires Python to be installed in order to run the binary. Scenario Manager can be run with the command SafeCOP_ScenarioManager.exe. Currently the manager does not take any parameters and the results are only visualised directly on the screen.

3.5 Inputs/Outputs

The framework is stand-alone. Scenarios can be defined with input from partners and then be simulated.

3.6 Use Cases Interactions

The framework is developed within UC5.2.

4. Bibliography

D5.4 Cooperative bathymetry demonstrator (MARO, DEM, CO, M33)