D2.4 DYNAMICS CP(H)SS AND COGNITIVE COMMUNICATIONS MODULES

Authors Wojciech Jaworski (ROBOTEC), Jordi Contreras (I2CAT), University of Peloponnese (UoP)

Work Package WP2

Abstract

This document that is the deliverable D2.4 "Dynamic CP(H)SS and Cognitive Communications Modules", represents the output of the Task 2.2 "CPS Inter and Intra Communication Models". It describes the protocols and simulation tools to be used for Inter and Intra Communications.

Funded by the Horizon 2020 Framework Programme
of the European Union

Deliverable Information

Work Package	WP2
Task	T2.2
Deliverable title	D2.4: Dynamic CP(H)Ss and Cognitive Communications Modules
Dissemination Level	Public
Status	F: Final
Version Number	1.0
Due date	M18

Project Information

Project start and duration	01/01/2017 – 31/12/2018, 24 months
Project Coordinator	Industrial Systems Institute, ATHENA Research and Innovation Center 26504, Rio-Patras, Greece
Partners	1. ATHINA-EREVNITIKO KENTRO KAINOTOMIAS STIS TECHNOLOGIES TIS PLIROFORIAS, TON EPIKOINONION KAI TIS GNOSIS (ISI) the Coordinator 2. FUNDACIO PRIVADA I2CAT, INTERNET I INNOVACIO DIGITAL A CATALUNYA (I2CAT), 3. IBM ISRAEL - SCIENCE AND TECHNOLOGY LTD (IBM ISRAEL 4. ATOS SPAIN SA (ATOS), 5. PANASONIC AUTOMOTIVE SYSTEMS EUROPE GMBH (PASEU) 6. EIGHT BELLS LTD (8BELLS) 7. UNIVERSITA DELLA SVIZZERA ITALIANA (USI), 8. TAMPEREEN KORKEAKOULUSAATIO SR (TAU) 9. UNIVERSITY OF PELOPONNESE (UoP) 10. CATALINK LIMITED (CATALINK) 11. ROBOTEC.AI SPOLKA Z OGRANICZONA ODPOWIEDZIALNOSCIA (RTC) 12. CENTRO RICERCHE FIAT SCPA (CRF) 13. PANEPISTIMIO PATRON (UPAT)
Website	www.cpsosaware.eu

Control Sheet

VERSION	DATE	SUMMARY OF CHANGES	AUTHOR
0.1	10/5/21	*ToC*	*Jordi Contreras, I2CAT*
0.3	26/5/21	Intra and Inter Comms first contribution	*Jordi Contreras, I2CAT* *Wojciech Jaworski, ROBOTEC* *Konstantinos Antonopoulos, University of Peloponnese*
0.4	3/6/21	Intra and Inter Comms chapters	*Wojciech Jaworski, ROBOTEC* *Konstantinos Antonopoulos, University of Peloponnese*
0.9	7/6/21	Internal review, draft submission	*Jordi Contreras I2CAT*
1.0	18/6/21	*Final version*	*Jordi Contreras, I2CAT* *Wojciech Jaworski, ROBOTEC* *Konstantinos Antonopoulos, University of Peloponnese*

	NAME
Prepared by	I2CAT
Reviewed by	Isidoros Michalellis, UPAT Petros Kapsalas, Panasonic
Authorised by	ISI

DATE	RECIPIENT
18/06/2021	Project Consortium
23/06/2021	European Commission

Table of contents

Tables

Figures

Acronyms

OBU	On-Board Unit
BLE	Bluetooth Low Energy
V2X	Vehicle-to-Anything
OSI	Open System Inteconnection
ITS-S	Intelligent Transport Systems Station
BSA	Basic Set of Applications
RSU	Roadside Unit
CAM	Cooperative Awareness Message
DENM	Decentralized Environmental Notification Message
MAP	Map Message
SPAT	Signal Phase and Timing Message
SAM	Service Announcement Message

1 Executive Summary

This deliverable, D2.4 Dynamics CP(H)Ss and cognitive communications modules, describes the Inter and Intra Communication models for the CPSoSaware Project. It defines both models, providing the foundation of technologies and simulation tools used. The intra communications model is based on NS3 simulator, while the inter communication model V2X concerns the communication between the different CPSos devices.

2 Introduction

Deliverable D2.4 contains the output of the Task T2.2, CPS Inter and Intra Communication Models.

In this document, different communication models provided by network simulators for both inter and intra communication layer of the CPSoSaware architecture, are described. The goal of this deliverable is the analysis and evaluation of these communication models and a high-level categorization of them. This categorization will drive Task T4.4, CPSoS Simulation Tools and Integration, where the simulation-based network communication performance optimization process will be accomplished, driven by different communication scenarios.

3 Structure of Document

3.1 Purpose and scope

This deliverable explores the different technologies available for CPS Intra and Intercommunication models and establishes a set of tools and simulators which will be used as basis for analysis in WP4 and later on, used in real scenarios in WP6. So, the scope of the deliverable is to introduce the technologies and tools used afterwards.

3.2 Approach

Section 4 describes the CPS Intra and Inter communication models, the challenges and the technologies selected to be evaluated.

Section 5, devoted to the Intra Comm Model is structured as follows. Firstly, a high-level analysis of the selected network simulator is presented. Next, auxiliary models provided by NS3 are analysed, mainly modelling the physical environment. After the auxiliary models, we analyse and categorize the communication models on which we are going to focus in the CPSoSaware project. Finally, we derive critical network parameters, and evaluate them under simple communication scenarios.

Finally, Section 6 is focused on the Inter Comm Model, details how the V2X model is implemented, and the simulation tools used. Two approaches have been followed, one based on open-source tools, which focuses on the radio communication simulation and a proprietary one more oriented towards high level cooperative awareness.

4 Description of CPS Inter and Intra Communication Models

4.1 Intra Comms

Efficient communication in CPS resource-constrained application posing demanding requirements comprise a challenging area calling for both new protocols and an efficient, cooperative communication primarily, focusing on low power communication. Another challenge posed by IoT applications, is that there is no unique solution that will address typical functional and non-functional requirements such as time-constrain communications, security, and scalability. Optimum selection, configuration and deployment also remains challenging due to orthogonal network requirements and resource limitations. A common problem, here, is the cost related to deploying and running real-world mobile testbeds, to evaluate different network configurations. Taking this into account, the role of the Intra-communication layer in the CPSoSaware, is to orchestrate the modelling procedure of low power, short-distance communication technologies with respect to different use case scenarios, as close as to the real-world testbeds, and try to find an optimal network configuration, that full-fill functional and non-functional requirements. In that respect, different prominent personal-area communication technologies, such as BLE, Zigbee, and WiFi will be explored under different scenarios, that will drive the optimal communication technology and its configuration. Finally, the Intra-communication layer will provide proper modeling interfaces for coupling virtual system prototyping platforms with a network simulation platform, which will generate development feedback, and drive real network deployments, based on specific application requirements.

The main contribution on this deliverable is to analyse the models provided by the simulation environment. Firstly, we split the analysis of models in two sections, namely, auxiliary and communication models. Auxiliary models section describes the communication environment (e.g., the mobility of the nodes), and mainly used to represent different use case scenarios. The communication models section, referring to the communication technologies that we target to evaluate. According to the analysis, we characterize critical network parameters on both auxiliary and communication models, provided by the simulator, which will be our guide to the network performance optimization process.

4.2 Inter Comms

V2X Communications have been selected in the CPSoSaware Project to perform the communications with the network elements and perform the commissioning. An onboard unit (OBU) will be mounted in vehicles, connected to the Roadside units (RSUs) which will provide access to the Edge functionality.

4.2.1 V2X Protocol Architecture

The C-ITS network architecture consists of different entities, or ITS stations (ITS-Ss), communicating with each other. These are:

- Personal ITS-S – handheld devices of pedestrians
- Vehicle ITS-S – OBU mounted on vehicles
- Central ITS-S – traffic management centres
- Roadside ITS-S – RSU or fixed traffic infrastructures

The combination of any of these entities results in different communication modes.

The ITS-S reference architecture defines the protocol stack implemented on each station. It comprises four horizontal layers along with two vertical entities [1]. It is analogous to the OSI model, except that it extends the model to include the ITS applications.

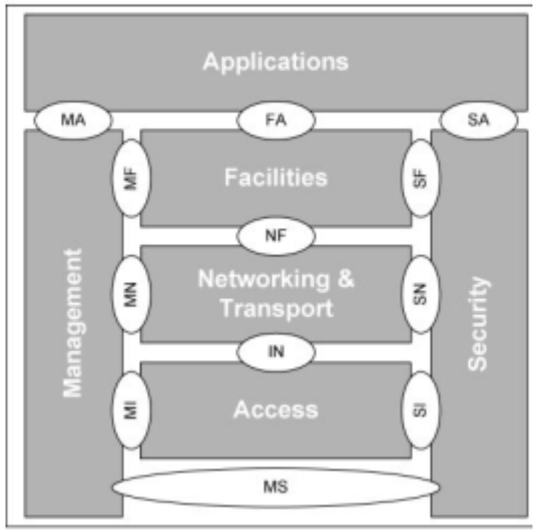


Figure 1 - ITS-S reference architecture

ITS-S Reference Architecture	OSI Model
Applications	-
Facilities	Application
	Presentation
	Session
Networking and Transport	Transport
	Network
Access	Data Link
	Physical

Table 1 - Mapping between ITS-S reference architecture and OSI model

4.2.2 Applications

ITS applications are formed by complementary ITS-S applications. A group of applications and use cases is known as the Basic Set of Applications (BSA). These use cases are categorized into the following three classes [2]:

1. Active road safety: The goal of this class is to improve traffic safety by preventing road casualties.

2. Cooperative traffic efficiency: The goal of this class is to improve road traffic management, and increase traffic efficiency in terms of travel times, fuel consumption, emissions, etc.
3. Other applications: These include applications providing other services such as those for infotainment.

Applications class	Application	Use Cases
Active road safety	Driving assistance – Co-operative Awareness (CA)	Emergency vehicle warning Slow vehicle indication Intersection collision warning Motorcycle approaching indication
	Driving Assistance – Road Hazard Warning (RHW)	Emergency electronic brake lights Wrong way driving warning Stationary vehicle –accident Stationary vehicle – vehicle problem Traffic condition warning Roadwork warning Collision risk warning Decentralized floating car data – Hazardous location Decentralized floating car data – Precipitations Decentralized floating car data – Road adhesion Decentralized floating car data – Visibility Decentralized floating car data – Wind
Co-operative traffic efficiency	Speed Management (CSM)	Regulatory/contextual speed limits notification Traffic light optimal speed advisory
	Co-operative Navigation (CoNa)	Traffic information and recommended itinerary Enhanced route guidance and navigation Limited access warning and detour notification In-vehicle signage
Co-operative local services	Location Based Services (LBS)	Point of Interest notification Automatic access control and parking management ITS local electronic commerce Media downloading
Global internet services	Communities Services (ComS)	Insurance and financial services Fleet management Loading zone management
	ITS station Life Cycle Management (LCM)	Vehicle software/data provisioning and update Vehicle and RSU data calibration

Table 2 - Basic set of applications

4.2.3 Facilities

The ITS facilities layer maps to layers 5, 6 and 7 of the OSI reference model. As such, it exhibits the corresponding functionalities of those three layers combined with ITS-specific ones. Its key role is to provide service to the ITS applications in the upper layer, and thus, the facilities are also referred to as basic service. Some of the facilities are listed in Table X, and can be grouped in two ways, according to: type of support and scope of support provided to the ITS BSA[2].

Classification	Facility name	Brief description
Common facilities for the application support	Priority management	Message and use case priority assignment.
	Identities management	Manage the station identifier used by the applications and the V2X messages.
	HMI interface	Provide common interface to multiple HMIs.
	CAM management	Provide management support for Cooperative Awareness Message.
	Security access management	Provide and manage the high layer security requirements and data to the security entity.
	Time management	Provide the time management and time synchronization service within the ITS station.
	Service management	Manage the supporting ITS service and applications within the ITS station.
Common facilities for the information support	Station type/capability	Manage the ITS station type and capabilities information.
	Position management	Provide and manage the station position and movement information.
	Location management	Provide location referencing functionalities for the station positioning according to the application requirements.
	Data presentation	Provide presentation support for the V2X messages.
Common facilities for the communication support	Communication management	Contribute from the high layer for the management and the selection of the optimal communication profiles to be used for the V2X message transmission.
	Addressing mode	Select the addressing mode for the V2X message transmission and provide the message dissemination requirements to the network and transport layer.
Domain facilities for the application support	Mobile station dynamics	Manage the vehicle ITS station dynamics information from the in-vehicle networks and vehicle electronic functions.
	Mobile station status monitoring	Monitor mobile station status from in-vehicle network and vehicle electronic functions and provide information for applications.
	DENM management	Manage DENM and DENM protocol.
	Roadside ITS station state monitoring	Monitors the roadside ITS station status.
	Client ID management	Manage and define the service clients profile information.
	Web service	High layer protocols for the web service e.g., SOA application protocol support.
	Billing and payment	Provide service access to the billing and payment service.
	GIS support	Provide the interface to the GIS service.
	Discovery mechanism	Discover the user of a community service either by a service announcement (passive) or by a subscription (active).
	Station life cycle	Provide the support for station software updating and data updating.
	Relevance check	Provide the relevance check for the received information from other ITS stations, according to the application requirements.
Domain facilities for the information support	LDM	LDM database
	Map data base	Provide interface to the map database at the central ITS station.
	Service content database	Manage a database of the ITS service content.
	RSU registration	Manage the roadside ITS stations and their information that are under the control of a central ITS station.

	User repository	Management of the user information at a central ITS station providing an ITs service.
	Fleet Monitoring	Monitor the community service behaviour at the central ITS station relevant.
	Message queuing	Manage the V2X messages queuing based on the message priority and the client services/use case requirements.
Domain facilities for the communication support	Session support	Support the communication session establishment and closure.

Table 3 - List of ITS facilities

4.2.4 Networking and Transport

The Basic Transport Protocol (BTP) provides an end-to-end, unreliable, and connectionless transport service. It is responsible for multiplexing the messages from the different processes at the ITS facilities layer, and at the other end, demultiplexing of messages received through the GeoNetworking protocol. The way multiplexing/demultiplexing works is based on ports, which act as identifiers to distinguish different processes running on the ITS station. Moreover, BTP allows the facilities layer to access the services provided by the GeoNetworking protocol, as well as the exchange of protocol control information between those two entities [3].

There are two types of BTP headers, which are indicated in the Next Header (NH) field of the GeoNetworking Common header. BTP-A is for interactive packet transport, while BTP-B signals the non-interactive. Moreover, they differ in packet structure, with BTP-A containing both: source and destinations ports, and BTP-B specifying only the destination port with the addition of destination port information in case of well-known ports, making clear that the BTP-B is designed for broadcast communications. Well known ports are listed in Table 4.

Well known BTP port	ITS facilities layer entity
2001	CAM – Cooperative Awareness Message
2002	DENM – Decentralized Environmental Notification Message
2003	MAP – Map message
2004	SPAT – Signal Phase and Timing
2005	SAM – Service Announcement Message

Table 4 - BTP ports

The GeoNetworking protocol is a network-layer protocol that uses geographical positions and areas to route packets across the ITS ad hoc network. It enables infrastructure-less communication, and meets the vehicle networking requirements, such as support for high node mobility and continuously changing network topology [4].

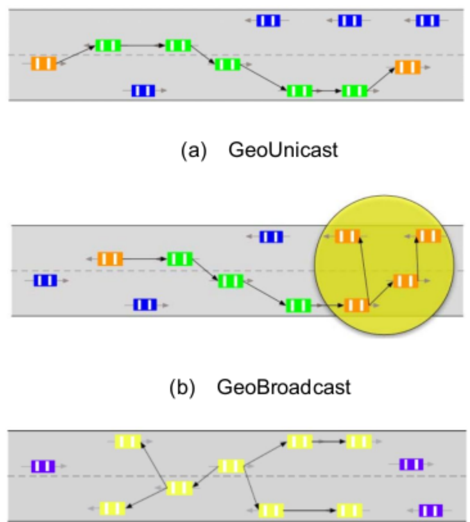
The GeoNetworking protocol has the following main functions:

1. Geographical addressing: A packet is sent to a destination node with a specific geographical position or to several destination nodes belonging to a geographical area.

2. Geographical forwarding: Each node maintains knowledge of the network topology. When a node receives a packet, it examines the destination field, and compares the indicated geographical address to its knowledge of the network topology to make forwarding decisions.

The GeoNetworking routing employs different packet forwarding schemes, as shown in Figure 2:

1. GeoUnicast: The packet is continuously forwarded by intermediate nodes until it reaches its destination.
2. GeoBroadcast: The packet is continuously forwarded until it reaches its destination geographical area. The nodes inside the area re-broadcast the packet, unlike the GeoAnycast.
3. Topologically-scoped broadcast: The packet is continuously re-forwarded until the n-hop node.



(a) GeoUnicast

(b) GeoBroadcast

(c) Topologically-scoped broadcast

Figure 2 - GeoNetworking routing schemes

4.2.5 Access

The ITS access layer maps to the data link and physical layers of the OSI reference model. The data link layer consists of the MAC and the Logical Link Control (LLC) sublayers. The ITS access layer technology is termed ITS-G5, which is based on the IEEE 802.11 Wireless Local Area Network (WLAN) standard. IEEE 802.11p corresponds to the PHY and MAC layers and is an enhancement of IEEE 802.11a.

IEEE 802.11p employs an almost identical physical layer as the IEEE 802.11a. However, some differences are needed to be introduced for it to be able to handle the high node mobility and steadily changing vehicular environments. For one, IEEE 802.11p utilizes the 10 MHz frequency channel bandwidth and works on the 5.9 GHz band, as opposed to the 20 MHz of IEEE 802.11a and 5 GHz band, to make the signal more robust to fading and other propagation effects.

ITS-G5 frequencies are allocated depending on their purpose of use, which also differ on performance requirements. To enable various ITS applications, one control channel (CCH) and seven services channels (SSH) are allocated [5].

IEEE 802.11p uses a MAC algorithm known as the Enhanced Distributed Coordination Access (EDCA). It works like the Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA) algorithm but allows the prioritization of data traffic. It defines separate queues corresponding to different access categories (ACs).

AC	TC ID	CW (min)	CW (max)	AIFS	Intended Use
AC_VO	0	3	7	58 µs	High priority DENM
AC_VI	1	7	15	71 µs	DENM
AC_BE	2	15	1023	110 µs	CAM
AC_BK	3	15	1023	149 µs	Multi-hop DENM, other data traffic

Table 5 - ITS-G5 Traffic classes

In an ITS ad hoc network, the network topology varies constantly, and the number of vehicles within the communication range is unpredictable. In the case of high-density scenarios, the communicating vehicles may require a number of resources beyond the channel capacity. As such, the Decentralized Congestion Control (DCC) mechanism is necessary to avoid channel congestion and allow a fairer access to the limited resources. The way DCC works is that the vehicle adapts its transmission parameters according to the measured channel load. The different DCC access techniques which are used to control the channel load are:

1. Transmit Power Control (TPC): adjust the output power to reduce the resulting interference.
2. Transmit Rate Control (TRC): adjust the time between consecutive packets, such as is increased in high density scenarios.
3. Transmit Data rate Control (TDC): adjust the transfer rate, such that it is lowered at high load scenarios.

5 Intra Comm Model development

5.1 NS3 Simulation Environment

NS-3[1] is a discrete event network simulator, which has gained recognition and acceptance by the industry and the research community as a tool of choice for network performance and evaluation simulations. The simulator has a multi-layered framework, while each layer depends on its lower layers. A high-level architecture of the NS3 is shown in Figure 3, below.

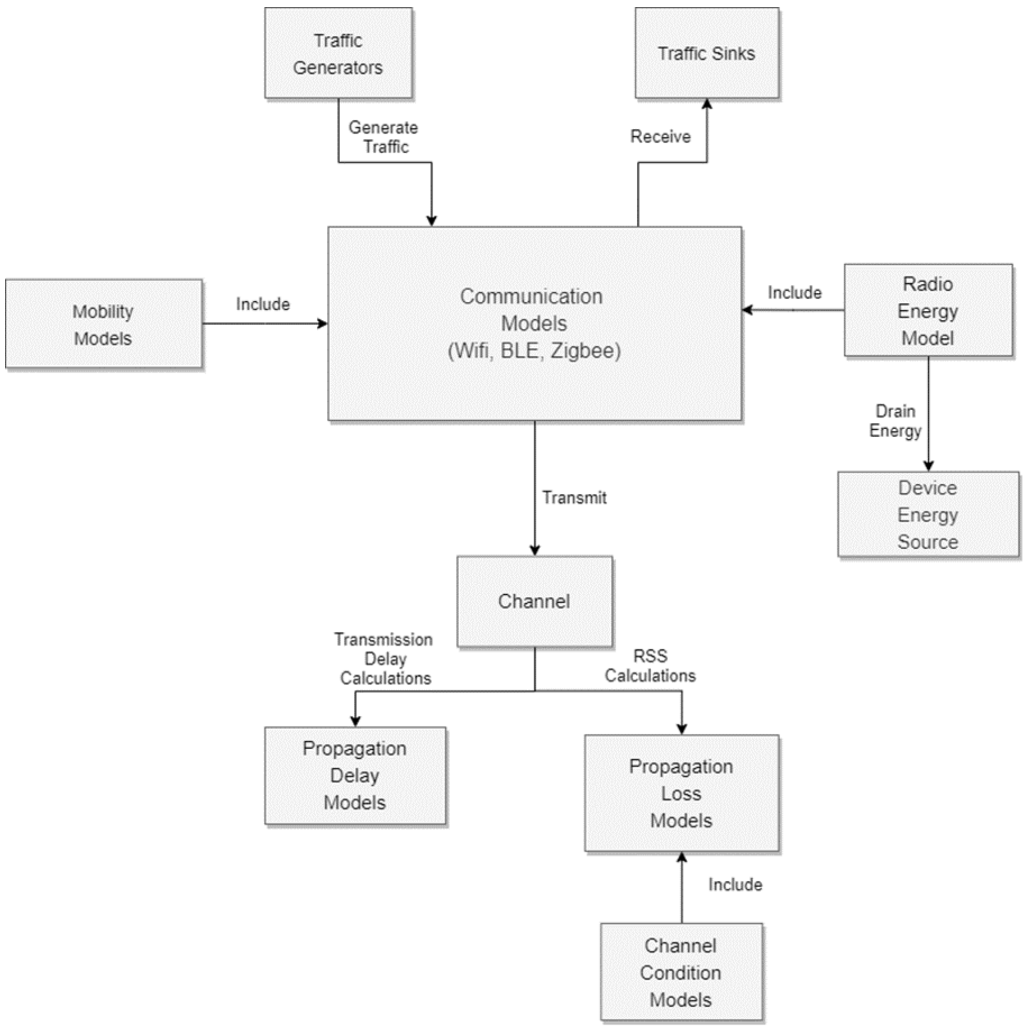


Figure 3 - NS3 Simulation Overall Architecture

Starting from the top, traffic generators/sinks modules are used to represent the application's specific scenario, allowing the importation of different types of traffic to our simulations. Mobility Models introduce the different movement patterns that simulation nodes can include to the overall simulation scenario. Radio energy models used to evaluate the energy consumption of the overall network or per-node power consumption. Finally different channel models are used to simulate different physical phenomena.

[1] https://www.nsnam.org/

5.2 Auxiliary Models

5.2.1 Traffic Model (Generator / Sinks)

One of the main parameters when simulating real-world scenarios is how much data is created and circulated in the network. Two major things that we need to differentiate, on the traffic models, is the traffic size and frequency. We can categorize these types of traffic into the following categories: constant traffic, uniform traffic and finally Poisson traffic. Poisson distribution traffic generation is close to traffic generated in real world environments.

5.2.2 Mobility Models

Mobility models are used to simulate and evaluate the performance of mobile wireless systems. The definition of realistic mobility models is one of the most critical and, at the same time, difficult aspects of the simulation of applications and systems designed for mobile environments. The mobility models are closely related with the propagation models. Table 6 shows different mobility models, included in NS3 simulator and their applicability, in use cases scenarios define in the CPSoSAware project.

Model	Use Cases
Constant Position Mobility Model	Industrial
Constant Velocity Mobility Model	Automotive
Hierarchical Mobility Model	Industrial, Automotive
Waypoint Mobility Model	Automotive

Table 6 - NS3 Mobility Models / Applicable Scenarios

5.2.3 Energy Models

Energy consumption is another key issue for wireless devices, and wireless network researchers often need to investigate the energy consumption at a node or in the overall network while running network simulations. NS-3 provides models for simulating energy consumption at node-level. The modelling is composed of two different models, the Device Energy Model, and the Energy Source.

The Device Energy Model is the energy consumption model of a device installed on the node. It is designed to be a state-based model where each device is assumed to have several states, and each state is associated with a power consumption value. Whenever the state of the device changes, the corresponding Device Energy Model notifies the Energy Source, which in turn, drains the corresponding current.

The Energy Source represents the power supply that is attached on each node. A node can have one or more energy sources, and each energy source can be connected to multiple device energy models.

Connecting an energy source to a device energy model implies that the corresponding device draws power from the source. Every time the Energy Source receives a notification from the Device Energy Model, it calculates the new total current draw and updates the remaining energy.

5.2.4 Propagation Models

Radio propagation models are used to simulate wireless signal attenuation in different environments, with obstacles or not, distance between nodes etc. NS3 split the propagation models in two categories, the *delay* and *loss* models. These models are attached to communication channels (NS3 models) and calculate the signal strength on the receiver side. NS3 provides specialized Channel models (E.g Wifi Channel Model) as well as more generic models like Spectrum Channel model, which can be used to simulate Spectrum Band channel models.

Delay models used to determine the packet delay. Delay models are simple, and in general the Constant Speed Propagation Delay Model is used, which calculates the delay based-on the distance between the transmitter and the receiver.

An important part of any wireless network is the appropriate choice of the Propagation Loss Model, which is required to compute the signal strength of a wireless transmission at the receiving stations. Propagation loss depends on the condition of environment (urban, rural, dense urban, suburban, open, forest, sea etc), operating frequency, atmospheric conditions, indoor/outdoor & the distance between the transmitter & receiver [6]. The factors that affect the signal attenuation are Spreading, Attenuation, Fading, Doppler Effect and Shadowing. Propagation models can be categorized in three groups, Abstract, Deterministic and Stochastic models [7]. Abstract models do not provide realistic results. Deterministic models, affected mainly by the distance between the sender and the receiver. Finally, the stochastic models are used in combination with deterministic models in order to provide non-deterministic results.

Table 7 shows propagation models included in the NS3 simulator and their applicability.

Model	Category	Applicable
FixedRSS Loss Model	Abstract	Everywhere
MatrixPropagation Loss Model	Abstract	Everywhere
RandomPropagation Loss Model	Abstract	Everywhere
FriisPropagation Loss Model	Deterministic	Line-Of-Sight Free Space Environment
LogDistancePropagation Loss Model	Deterministic	Suburban Scenarios
ThreeLogDistancePropagation Loss Model	Deterministic	Suburban Scenarios

TwoRayGroundPropagation Loss Model	Deterministic	Line-Of-Sight Free Space Environment
JakesPropagation Loss Model	Stochastic	Cellular Areas
Nakagami Propagation Loss Model	Stochastic	Variables because it used with conjunction with other propagation models
ItuR1238Propagation Loss Model	Deterministic	Short Range Outdoor Line-of-Sight
ItuR1411LosPropagation Loss Model	Deterministic	Short Range Outdoor Non-Line-of-Sight Over Rooftops
ItuR1411NlosOverRooftopPropagation Loss Model	Deterministic	Urban Suburban Open Areas Environments Frequency 200 up to 2.6GHz
ThreeGppRmaPropagation Loss Model	Stochastic	Vehicular Environments
ThreeGppV2vUrbanPropagation Loss Model	Stochastic	Vehicular Environments
ThreeGppIndoorOfficePropagation Loss Model	Stochastic	Vehicular Environments
ThreeGppUmiStreetCanyonPropagation Loss Model	Stochastic	Vehicular Environments
ThreeGppUmaPropagation Loss Model	Stochastic	Vehicular Environments

Table 7 - NS3 Propagation Loss / Applicable Scenarios

5.2.5 Error Rate Models

Error Rate models are attached to the physical layer of the network stack and used to simulate packet loss (decoding errors), based on the mode of operation of the communication medium, using modulation scheme, coding rate and other PHY layer communication parameters. NS3 provides both stochastic and deterministic models for modelling packet error on packet reception, as shown in the following Table 8.

Model	Type
Rate Error Model	Stochastic
Nist Error Rate Model	Stochastic
YANS Error Rate Model	Stochastic
List Error Model	Deterministic
Receive List Error Model	Deterministic
Burst Error Model	Deterministic
Table Based Error Rate Model	Deterministic
DSSS Error Rate Model	Deterministic

Table 8 - Error Rate Models

5.3 Models for the Intra-Communication CPSoSAware Scenarios

5.3.1 WiFi (IEEE 802.11)

IEEE 802.11 is a set of standards for wireless local area network (WLAN) developed by the IEEE LAN/MAN Standards Committee (IEEE 802) in the 2.4GHz and 5GHz public spectrum bands, which targets to provide in-building broadband coverage as well as outdoor coverage that can reach up to 1500 feet distance. Each 802.11x implementation operates in different bands and offers different data rates, covered range etc.

5.3.1.1 Model Architecture

The architecture of the model is shown in Figure 4 below.

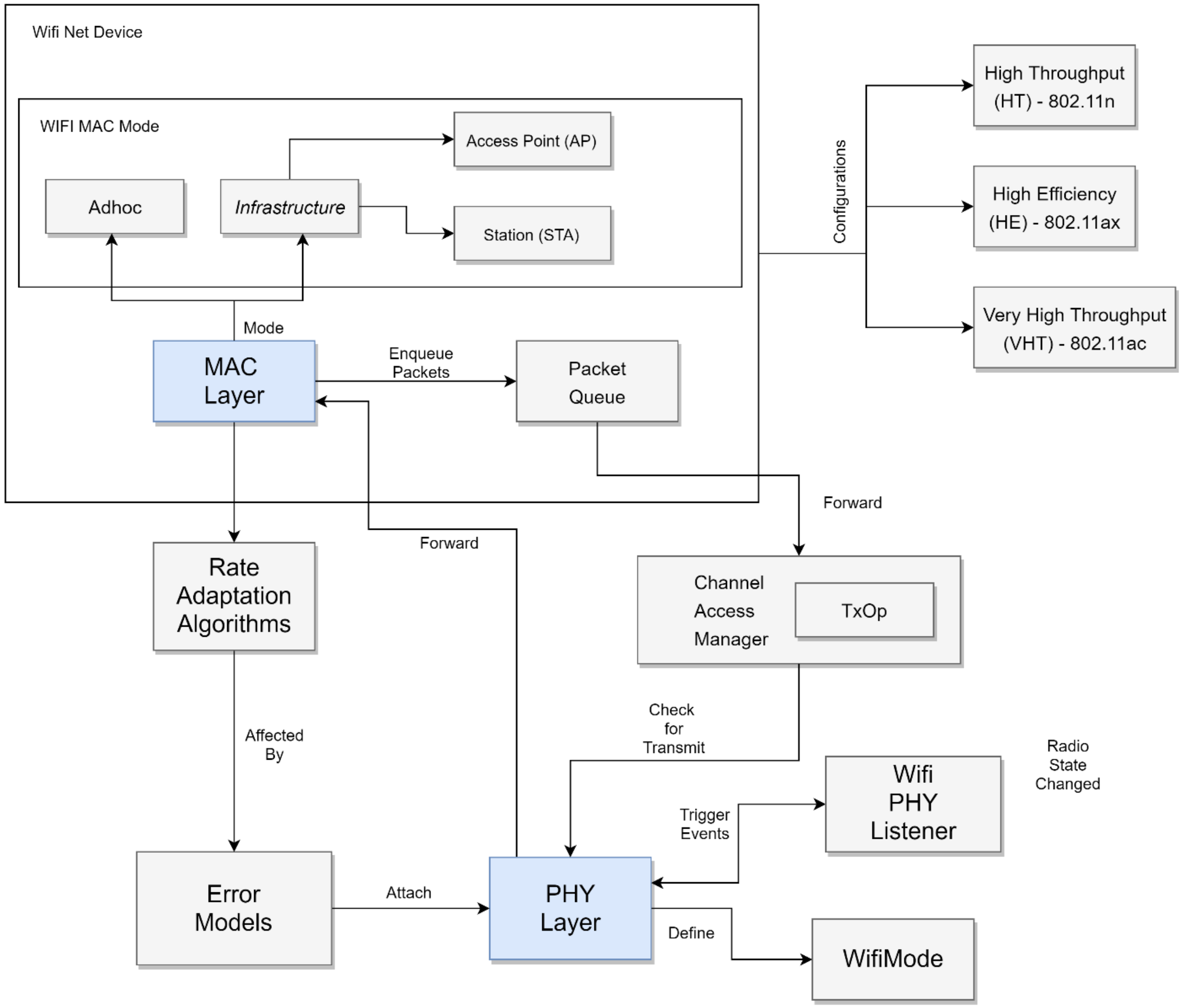


Figure 4 - WiFi Model Architecture

NS3 provides support for the following WiFi standards:

Frequency Band	Standard
2.4GHz	802.11a/b/g/n/ac/ax
5 GHz	802.11n/ac/ax
6 GHz	802.11ac/ax
5.9 GHz	802.11p

Table 9 - NS3 – 802.11x Support

A WiFi-enabled device in an NS3 simulation environment is modelled with WiFiNetDevice. The modules that comprise a WiFi device are the following.

- ***Channel,*** the communication channel of net-device, which allows the simulation of packet delay & loss, using different propagation models, see *Section 5.2.4*.
- ***Physical (PHY) Layer***, which is responsible for modelling the transmission/receptions of the packets. Also the energy consumption modelling is performed in this module.
- ***MAC Layer***, which is responsible to model the type of the WiFi device, with specific characteristics.
- ***Rate Adaptation Algorithm***, which is responsible to track the communication medium and accordingly to adjust the data rate, taking account packet reception parameters.

Physical Layer

The physical (PHY) layer models are responsible for modeling the transmission and reception of packets and tracking the energy consumption. There are mainly three evaluation points of the packet reception.

- Firstly, the evaluation of the packet reception is based on a probabilistic approach, where the probability of a successful packet reception depends on, the modulation scheme, on the signal to noise (SNR) (and interference) (SINR) ratio for the packet, and on the state of the physical layer.
- In a interference communication scenario, all the received signals so that the correct interference power for each packet can be computed when a reception decision has to be made
- And finally, the packet reception is determined by one or more error models corresponding to the modulation and standards are used to look up probability of successful reception.

The Energy consumption is mainly determined, and the energy is drained from the energy due to any PHY state switch. WiFi states are shown in the following Table 10. The following states are low level PHY layer flags that represents if the network device transmitting/receiving data (BUSY, TX, RX), or is in an intermediate state (IDLE, CCA, SWITCHING, SLEEP), where neither send nor receive packets.

PHY States
IDLE, CCA (Clear Channel Assessment), BUSY, TX (Transmitting), RX(Receiving), SWITCHING, SLEEP, OFF

Table 10 - WiFi Phy State.

MAC Layer

The MAC layer models define the mode of the WiFI network, which can be configured either to infrastructure or Ad-Hoc. In the case of infrastructure mode the nodes can be configured to act as Access Point (AP) or Station (STA).

MAC-high layer: This sub-layer is responsible for high level MACmanagement functions like probing, Association/ De-association and beacon generation. There are currently three MAC-high models; ApWifiMac, StaWifiMac, and AdhocWifiMac. The former two represent an AP and Non-AP nodes, respectively. The latter is the simplest of these models and is used in an IBSS network (ad hoc network). All the three models share a common parent, RegularWifiMac, which encapsulates all MAC functionality.

MAC Layer is generally divided in two sub-modules. The first one takes cares of RTS/CTS/DATA/ACK transactions. DCF and EDCAF functions are, also, implemented in this layer regarding Non-QoS and QoS amendments in WLANs. Another responsibility of the MAC layer is to handle packet queues, packet aggregation, fragmentation or packet retransmissions when QoS is disabled or supported, respectively.

.

Rate Adaptation Manager

IEEE 802.11 standard introduced multi-rate support, since then, a lot of research has been done on rate adaptation, dealing with the different parameters that lead to an estimation of the channel conditions and the metrics that affect the network performance [12]. Rate Adaptation is one of the key aspects of the functionalities of IEEE 802.11's physical layer. It works by assessing the channel conditions and taking a decision to adapt the transmission rate or transmission power by selecting a combination of transmission features, such as the modulation and coding schemes (MCS), guard interval, and channel width. NS3 provides many Rate-Adaptation Algorithms, as follows.

Algorithm	Rate Adaptation	Applicable
ArfWifiManager	Rate	802.11a/b/g
AarfWifiManager	Rate	802.11a/b/g
OnoeWifiManager	Rate	802.11a/b/g
CaraWifiManager	Rate	802.11a/b/g
RraaWifiManager	Rate	802.11a/b/g
IdealWifiManager	Rate	802.11a/b/g
MinstrelWifiManager	Rate	802.11a/b/g
MinstrelHtWifiManager	Rate	802.11n/ac/ax
AmrrWifiManager	Rate	802.11a/b/g
AarfcdWifiManager	Rate	802.11a/b/g

AparfWifiManager	Rate Transmission Power	802.11a/b/g
ParfWifiManager	Rate Transmission Power	802.11a/b/g
RrpaaWifiManager	Rate	802.11a/b/g
ConstantRateWifiManager	None	ALL

Table 11 - NS3 – Rate Adaptation Algorithms

5.3.2 802.15.4 (Zigbee)

IEEE 802.15.4 [8] is an important standard for wireless sensor networks and Internet of Things (IoT) applications. The standard specifies the physical layer and media access control for low-rate wireless personal area networks (LR-WPANs), and is the underlying protocol used in the majority of sensor network industrial deployments. ZigBee, in turn, builds on top of IEEE 802.15.4 protocol functionality by adding capabilities for more flexible network topologies, intelligent message routing and enhanced security measures.

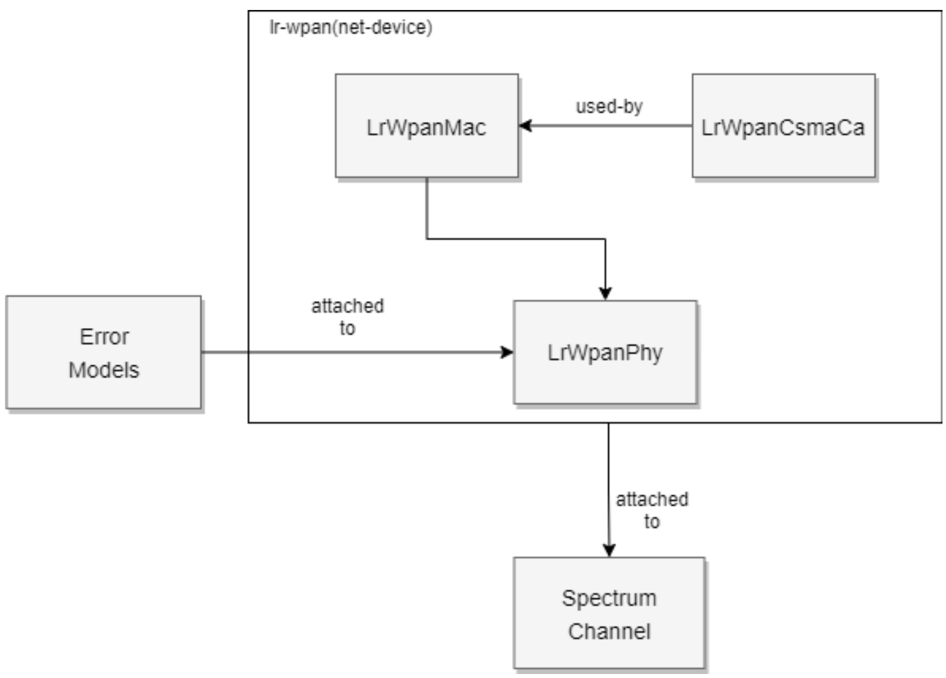


Figure 5 - 802.15.4 - Model Architecture

A Zigbee-enabled device in NS3 simulation environment is modelled with LrWpanNetDevice. The modules that comprise a Zigbee device are the following.

- ***Channel,*** the communication channel of net-device, which allows the simulation of packet delay & loss, using different propagation models, see *Section 5.2.4*. Here in contrast to WiFi modules only Spectrum Channels models are used.
- ***Physical (PHY) Layer** is* responsible for the transmission and reception of data. It dictates the radio bands to be employed and type of spreading and modulation techniques to be used. The modulation technique used is DSSS. The Physical layer in NS3 is provided by the *LrWpanPhy*, which is responsible for modelling the transmission/receptions of the packets. Also the energy consumption modelling is performed in this module, according to physical state transitions.
- ***MAC Layer*** is responsible for managing beacon transmission, access to channel, and association/ disassociation on the network. NS3 provides two models for the MAC layer, *LrWpanMac* the high level MAC implementation and the *Channel Access manager (LwWpanCsmaCa)*.

The Energy consumption is mainly determined, and the energy is drained from the energy due to the PHY state switch. Zigbee states provided by model shown in the following table,

PHY State
IEEE_802_15_4_PHY_BUSY, IEEE_802_15_4_PHY_BUSY_RX (Receiving), IEEE_802_15_4_PHY_BUSY_TX (Transmitting), IEEE_802_15_4_PHY_FORCE_TRX_OFF, IEEE_802_15_4_PHY_IDLE, IEEE_802_15_4_PHY_RX_ON, IEEE_802_15_4_PHY_SUCCESS, IEEE_802_15_4_PHY_TRX_OFF, IEEE_802_15_4_PHY_TX_ON

Table 12 - 802.15.4 PHY States

5.3.3 BLE (Bluetooth Low Energy)

Bluetooth Low Energy (BLE) is an innovative technology, developed by the Bluetooth Special Interest Group (SIG), which aims to become the best alternative to the huge number of standard wireless technologies. NS3 Simulator does not provide official models for BLE network. BLE protocol stack structured in three main blocks. The *Application layer* (App) is the highest block of the stack and represent the interaction with the user. The *Host* layer, is responsible for handling the connection mode of the device, end resides between the application layer and physical layer. Finally, at the bottom layer is the *Controller* which is responsible for transmission & packet reception.

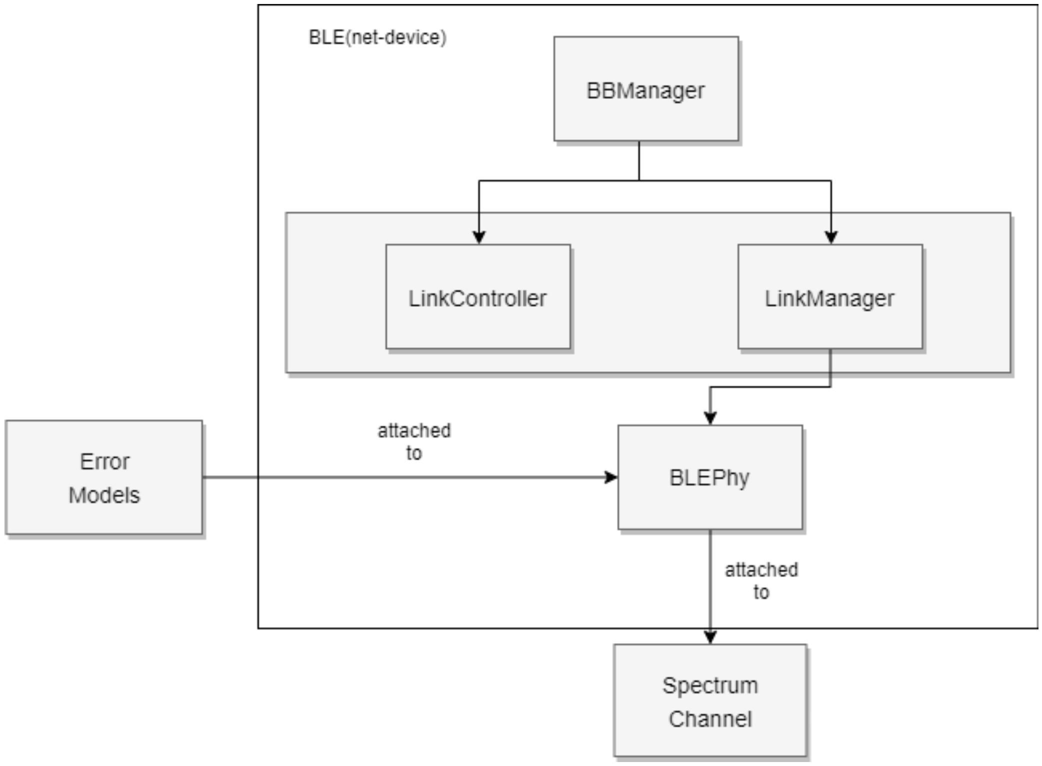


Figure 6 - BLE - Model Architecture

A BLE-enabled device in an NS3 simulation environment is modelled with BLE-Net-Device. The modules that comprise a BLE device are the following.

- ***Channel,*** the communication channel of net-device, which allows the simulation of packet delay & loss, using different propagation models, see *Section 5.2.4*. Here in contrast to WiFi modules only Spectrum Channels models can be used.
- ***Physical (PHY) Layer,*** implemented by *BLEPhy*, which is responsible for modelling the transmission/receptions of the packets. Also the energy consumption modelling is performed in this module, according to physical state transitions.
- ***Controller Layer,*** is responsible for the establishment and maintenance of the connection between the nodes. NS3 modules controller layer with *LinkController* and *LinkManager*.
- ***Host Laye***r, finally the host layer is represented using the BBManager.

The Energy consumption is mainly determined, and the energy is drained from the energy due to PHY states switch. BLE states provided by model shown in Table 13:

PHY State
IDLE, TX (Transmitting), TX_BUSY, RX(Receiving), RX_BUSY

Table 13 - BLE PHY States

5.4 Critical Network Parameters

In this section, we characterize critical network parameters, which affects the overall network performance, and present how these network parameters are modelled using NS3 simulator.

5.4.1 Simulation Environment

The main contribution here is to define general simulation parameters that affect network performance without taking into account the underline communication protocol. A high-level categorization of these parameters shown in Table 14 below.

Network Parameter	Description	Affects	Simulated By (NS3)
Topology	Network Topology, like Point-to-Point, Bus, Ring, Star	Throughput, Delay / Jitter	Network Devices
Radio	The switching between states that drain energy	Throughput, Delay, Power Consumption	Channel Models Radio Energy Models
Number of Nodes	Number of nodes that resides in the network. Affects the mode of each nodes, transmit, receive, transceiver	Throughput, Delay / Jitter	Network Devices
Mobility	The position of nodes inside the network area, as well as the different movement patterns that each node follows.	Throughput, Delay / Jitter	Mobility Models
Environment	The uniformity of the environment. Indoor/Outdoor areas, Urban etc..	Throughput, Delay / Jitter	Propagation Loss Models Propagation Delay Models
Communication Technologies Heterogeneity (Interference)	How many other different communication technologies exists on the same frequency band	Throughput, Delay / Jitter	Spectrum Channel Models Propagation Spectrum Loss Models
Packet Size	Per packet-size that nodes transmit over the air. The size of the packet is the summary size included from all layers of the stack (E.g application, MAC, PHY).	Throughput, Delay, Power Consumption	Traffic Generators Traffic Sinks MAC Layer Models PHY Layer Model

Table 14- Auxiliary Models – Critical Network Parameters

5.4.2 WiFi

A high-level of critical network parameters of WiFi shown in the Table 15 below.

Network Parameter	Description	Affects	Simulated By (NS3)
Standard	The communication standard at PHY Layer that is going to be used	Throughput, Power Consumption	PHY Layer Model.
Data Rate Adaptation	WiFi multi-data rate mechanism allows to adapt the data-rate according to error metrics	Throughput, Delay / Jitter Power Consumption	Rate Adaptation Models Propagation Loss Models Channel Conditions Models Error Rate Models.
Modulation & Coding Scheme	The combination of NSS, Channel Width, Guard Interval Width, Modulation and Coding Rate	Throughput Delay/Jitter	PHY Layer Model
Packet Aggregation	Packet Aggregation provided mainly from High Rate implementation of WiFi (HT, HE, VHT)	Throughput Delay / Jitter	MAC Layer Models PHY Layer Model
QoS	Enable/Disable Quality of Service in the communication	Throughput Power Consumption	MAC Layer Models
Communication Mode	The mode that network operators, Ad-Hoc or Infrastructure mode. According to the mode, there are three types of nodes, Access Point (AP), Station (STA), Ad-Hoc (AD)	Throughput, Delay / Jitter Power Consumption	MAC Layer Models
Channel Access Management	Configurations like Inter-Frame Spaces. These configurations affecting the transmission medium, and who has access to it. Packet Queue Management	Throughput, Delay / Jitter Power Consumption	MAC Layer Models PHY Layer Model

Table 15 - WiFi – Critical Network Parameters

5.4.3 802.15.4 (Zigbee)

A high-level of critical network parameters of 802.15.4 shown in the Table 16 below.

Network Parameter	Description	Affects	Simulated By (NS3)
Channel Access Management.	CSMA/CA parameters configuration for accessing the communication channel	Throughput, Power Consumption	MAC Layer Models
Beacon Probing	Beacon probing allows the network nodes to be synchronized with the coordination nodes. This feature can be enabled or not	Throughput Power Consumption	MAC Layer Models
QoS	Enable/Disable Quality of Service in the communication	Throughput Power Consumption	MAC Layer Models

Table 16 - Zigbee - Critical Network Parameters

5.4.4 BLE

A high-level of critical network parameters of BLE shown in the Table 17 below.

Network Parameter	Description	Affects	Simulated By (NS3)
Radio Communication Interval	This is a wide range of parameters which are related with the control messages interchanged between BLE enabled nodes	Throughput, Delay / Jitter Power Consumption	BLE-Phy Model
BLE Mode	BLE Modes: 1. Broadcasting 2. Scanning 3. Central 4. Peripheral	Throughput, Delay / Jitter Power Consumption	LinkController Model LinkManager Model

Table 17 - BLE Model – Network Parameters

5.5 Evaluation Results

In this section, we present a brief performance evaluation of the NS3 models. In the first two subsections, an evaluation of the auxiliary models is presented. Next, an evaluation of the chosen communication protocols is presented. Finally, in the last section, an explanatory demonstration is quoted, showing a joint job, that compromised by the way that we run the simulations, and their integration with external tools/systems.

5.5.1 Mobility Models Evaluation

A critical parameter that is affecting the overall architecture of the network, regardless of the communication technology, is the distance between the nodes, which is related also with the mobility of the installed nodes inside the simulation area. As shown, in Figure 7 below, as the distance between the nodes increases, also the packet loss is increasing. Other indications that can be observed by the graph, is that the number of network nodes is highly affecting the packet reception as the packet loss increases with the increase of the network nodes

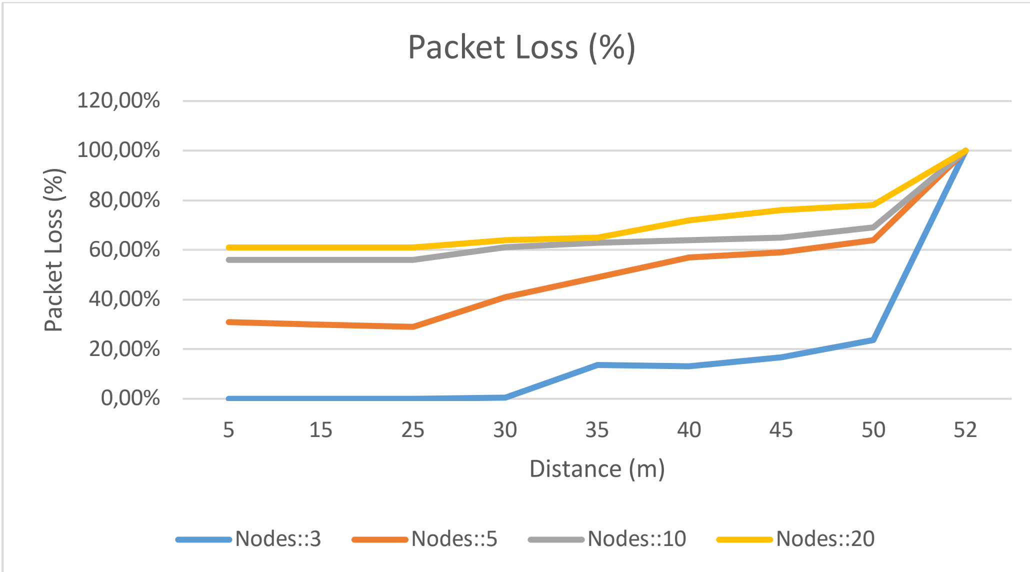


Figure 7 - Packet Loss (%)

5.5.2 Energy Model Evaluation

As described in the Energy models section, the total energy is consumed primarily on the radio state transitions. As depicted in Figure 8 below, optimizing radio state transitions can decrease energy consumption. On the other hand, increasing the packet size can result in the decrease of the power consumption.

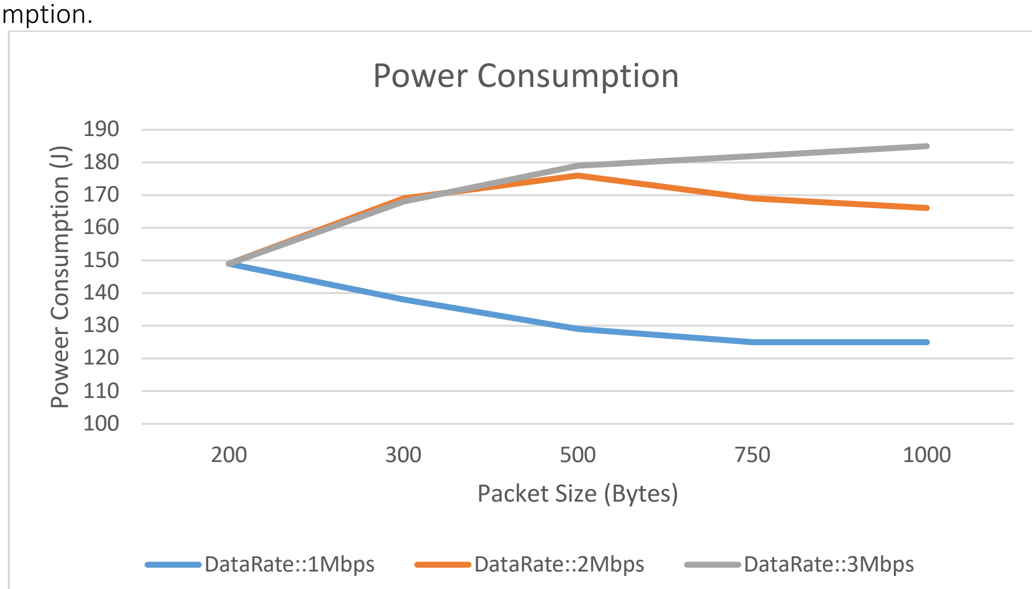


Figure 8 - Power Conpsumption

5.5.3 Communication Models Evaluation

In previous section, we present a basic evaluation of the auxiliary models, provided by NS3 simulator. In this section, we present an evaluation of the selected communication models, in the context of the

CPSoSAware, and demonstrate the results of high-level network parameters, that, as shown, affects the overall network performance.

5.5.3.1 WiFi Communication Model

As shown in the figures below, other critical parameters for WiFi that affect the network performance is application data in conjunction with packet payload. As shown in Figure 9, throughput is optimized according to the data and the packet size.

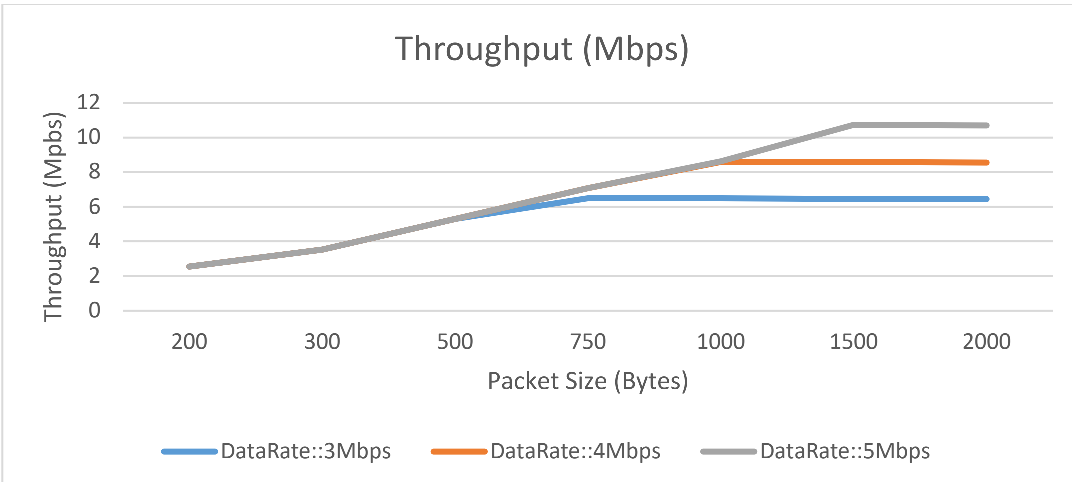


Figure 9 - Throughput

Figure 10 and Figure 11, also depicts how different data rates, combined with different packet payloads affects the network performance. As depicted the choice of the packet payload must be done in relation with the choice of the data rate, resulting to the minimization of the mean delay and packet loss.

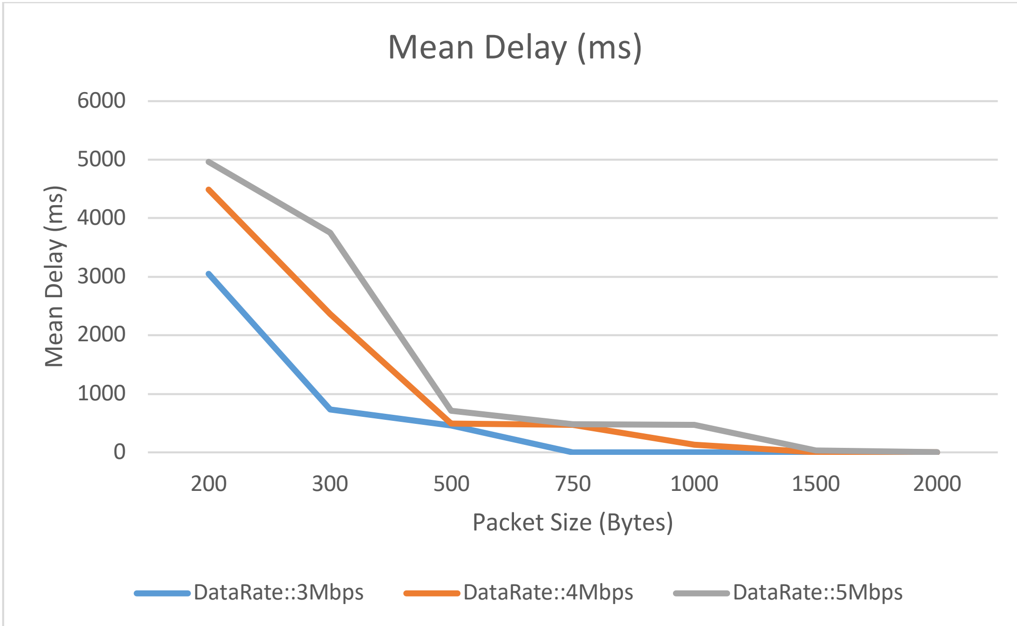


Figure 10 - Mean Delay

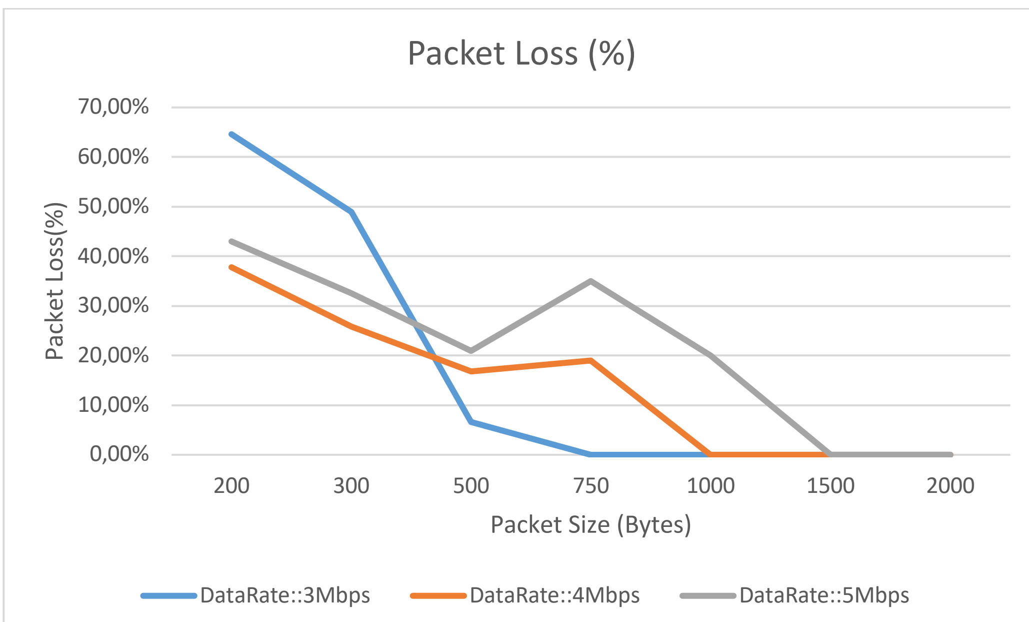


Figure 11 - Packet Loss

5.5.3.2 Spectrum Communication Model (802.15.4 (Zigbee) / BLE)

Another critical network parameter that affects the core network performance parameters is the PHY layer configuration scheme, which defines the upper transmissions limits. Figure 12 and Figure 13 , shows how the different modulation schemes affect the delay and throughput respectively.

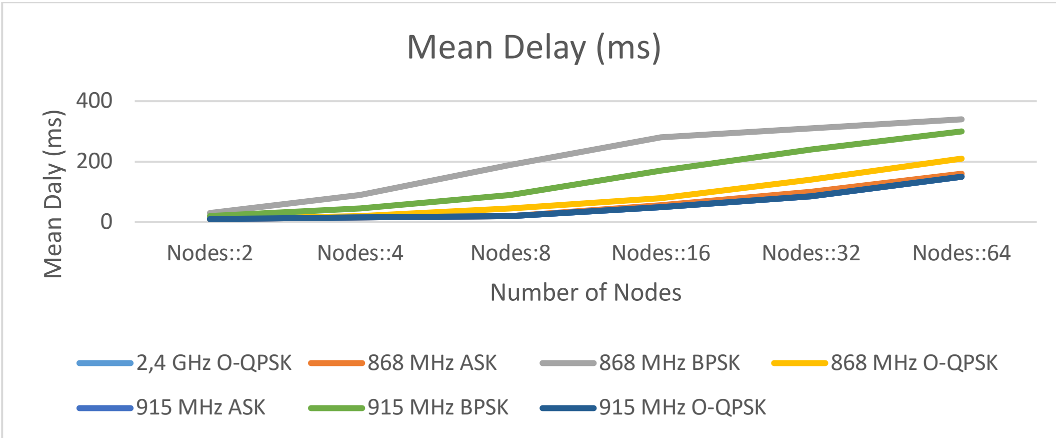


Figure 12 - Mean Delay

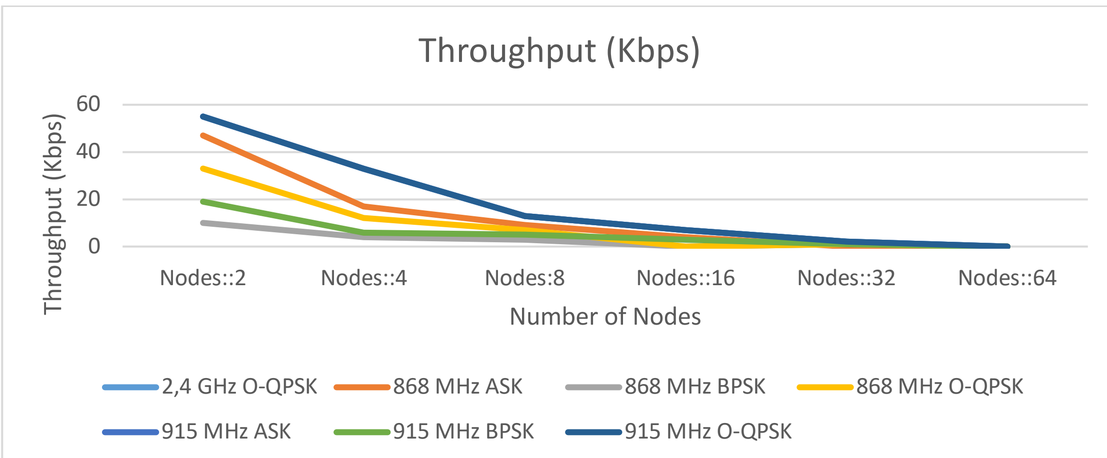


Figure 13 - Throughput

5.5.4 Evaluation Demonstration

In the following link (CPSOsAware-NS3 Simulator Integration Demonstration) you can watch a joint job that affects three CPSoSAware tasks (T2.2, T4.2, T4.4). A brief summary, of what you are going to view in this demonstration follows:

1. A simple Wifi simulation scenario in NS3, and its evaluation under two network parameters, Number of Nodes (General Network Parameter), and Rate Adaptation Algorithm (WiFi specific critical parameter). [***Task 2.2***]
2. Using the visualization graphs extracted from the simulation results, we can a determine an optimal combination of network parameters [***Task 4.2***].
3. The integration of the NS3 simulator with external tools, such as CI/CD infrastructures. [***Task 4.4***].

6 Inter communication model development

6.1 Simulation tools

For the development of the inter communications model a set of simulations will be executed and analysed to gather a better understanding of network behaviour in certain conditions. To do it we will use the help of simulators (OMNeT ++, SUMO) [9] and data analysis tools (SQLite, Pandas)[10][11].

6.1.1 Simulation Framework Overview

The IEEE 802.11p based simulator is composed of several simulation frameworks of different functionality.

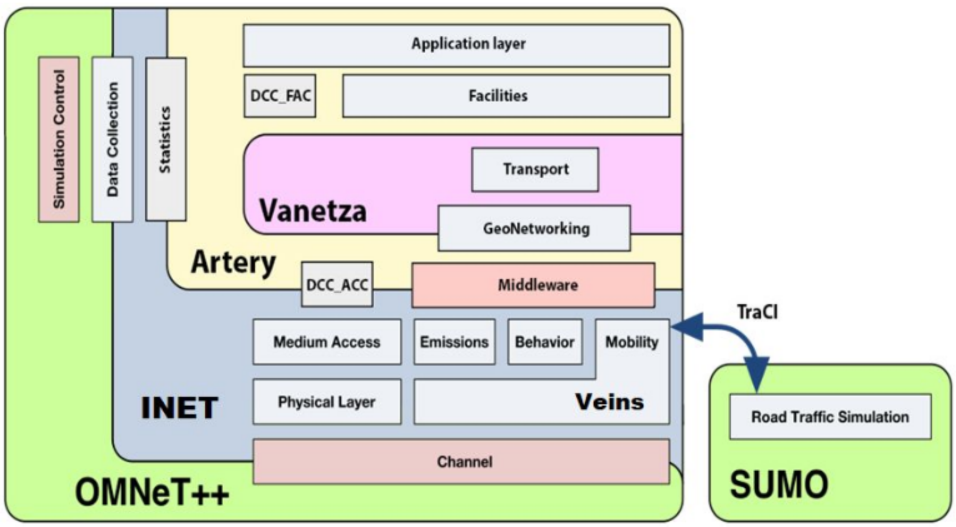


Figure 14 - Simulation Framework Overview

The Objective Modular Network Testbed in C++ (OMNeT++) is an extensible and modular simulation library and framework for the research and development of complex distributed systems. Countless simulation models and model frameworks have been written on top of OMNeT ++ by researchers in diverse areas, and vehicular networks is one of them. OMNeT++ works by assembling individual components/models to larger ones. This modularity makes it easy for the models to be reused and incorporated to different applications. Moreover, although OMNet++ is mainly used for building network simulators, it is also considered a network simulator platform by its growing number of users. Model frameworks are often used in conjunction with OMNeT++ to implement more specific functionalities.

The INET simulation framework is an open-source library containing various models to simulate communication networks, and in particular written for the OMNeT++ environment. Some of its features include models for the Internet stack (IPv4, IPv6, TCP, UDP) and wired/wireless interfaces (Ethernet, IEEE 802.11), and support for physical environment modelling (propagation model, presence of obstacles). Moreover, INET could be used as a base for creating other simulation frameworks.

Simulation of Urban Mobility (SUMO) is an open-source road traffic simulator. It allows the creation of different road topologies for the simulation, such as freeway and Manhattan grid scenarios, as well as the experimentation of various mobility models. Moreover, it is microscopic, as vehicles are individually modelled, and move independently through the network.

Artery was originally developed as an extension of Veins, although it could now be used independently. Artery corresponds to the application and facilities layers, which enable the generation of CAMs and DENMs. Moreover, Artery's middleware provides common facilities to the multiple ITS-G5 services running on individual vehicles. Figure 15 Shows the different components present in Artery, together with the corresponding configuration files:

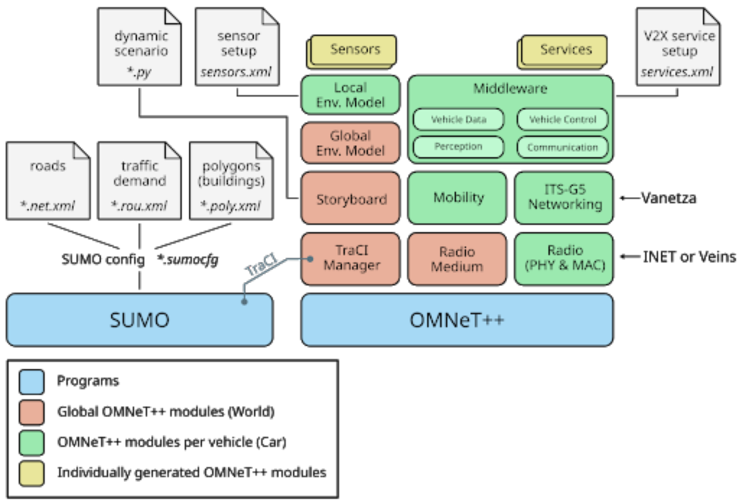


Figure 15 - Artery simulation Framework

Usually, two programs are running hand in hand when a simulation is running. On the left hand, there is the traffic simulator SUMO, and on the right-hand we have the OMNeT++ runtime environment. The interaction of these simulators is made possible using a TCP socket and a standardized protocol known as the Traffic Control Interface (TraCI). As such the movement of vehicles in SUMO is represented as the movement of nodes in OMNeT++.

6.1.2 Sumo Scenarios

For the performance of the simulations of this project two kinds of road topologies will be used and developed with the SUMO simulator: highway and a Manhattan grid.

6.1.2.1 Sumo Files

Each scenario is created using three SUMO files which are located in *artery/scenarios/i2cat*. They are the network file, the routes file and the configuration file.

The Network File contains the description of the physical topology of the scenario. This may include the roads, intersections, traffic logics and even roundabouts. Using the sumo naming convention, the roads or streets are referred to as edges, and the intersections as junctions or nodes. Two edges are connected by junctions.

The routes file specifies the vehicle types and routes for the vehicles in the simulation. The vehicle's type field includes the physical properties of the vehicle, such as shape, colour, maximum speed and minimum gap for the vehicle ahead. Different routes are identified by their *road id,* and each of them defines the relevant edges and direction of movement of vehicles. Moreover, a flow contains the information to control the vehicles inserted in the scenario and how they behave during the simulation.

The configuration file specifies the associated network and routes files for a given scenario. Moreover, it is possible to configure the *step-length,* which is the granularity of the simulation and has a minimum value of 1ms. It also corresponds to the time interval with which vehicle positions are updated.

6.1.2.2 Physical topologies

Different road topologies are used in this project. One of which is the highway scenario, which simulates direct line-of-sight (LOS) conditions and non-stop driving. The other one is the Manhattan grid scenario, which helps in understanding the effects of walls and buildings, as well as intersections. Moreover, the project defined a statistical region in the scenarios, highlighted in red below. Statistics are only recorded in this area to eliminate border effects. For instance, less vehicles may be present at either end of the highway scenario compared to its central region, and this consequently affects the carrier sense mechanism employed by IEEE 802.11p.

6.1.2.2.1 Highway Scenario

The grid scenario measured 3000m x 25m, with the statistical area bounded by 500m < x < 2500m, which corresponds to the central region of the highway as shown in Figure 16.

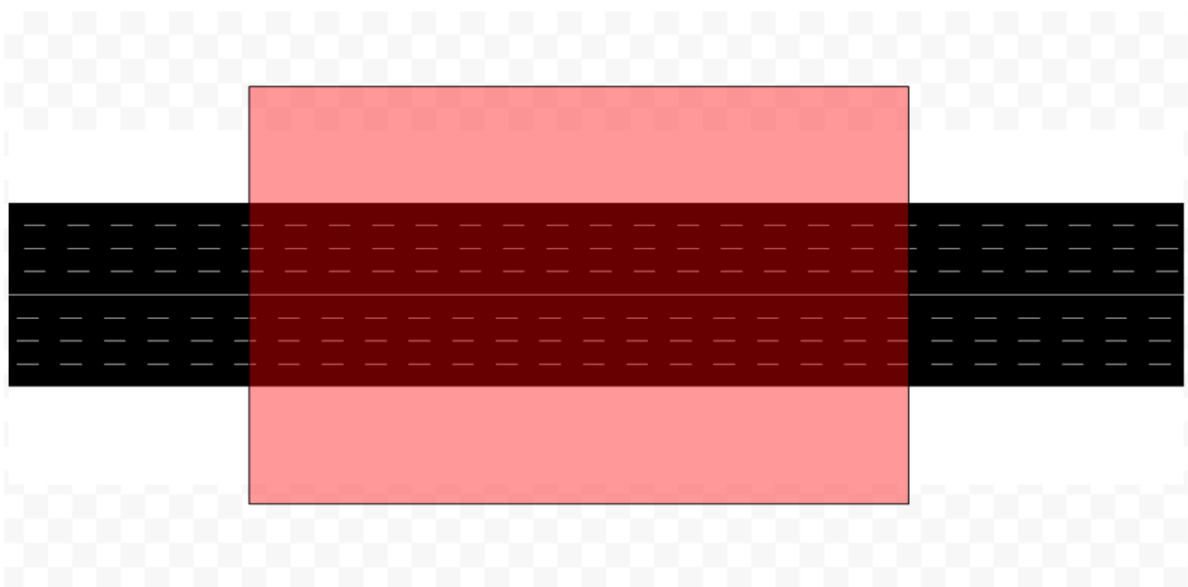


Figure 16 - Highway scenario

6.1.2.2.2 Manhattan grid Scenario

The Manhattan grid scenario measured 800m x 800m, with the statistical area bounded by 200m < x < 600m and 200m < y < 600m, which corresponds to the central region of the grid. The presence of walls could also be configured in *omnetpp.ini*.

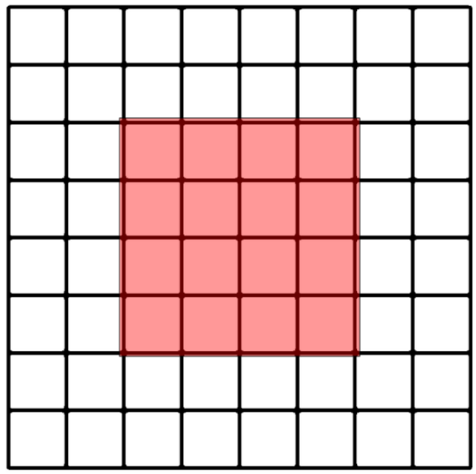


Figure 17 - Manhattan grid scenario

6.1.3 Simulation Parameters

Table 18 provides a summary of the parameters used in the simulations:

Category	Parameter	Value
Node	Operation mode	802.11p
	Carrier frequency	5.9 GHz
	Bandwidth	10 MHz
	Channel number	180
	Modulation	BPSK
	Bitrate	6 Mbps
	Transmitter power	200 mW
	Receiver sensitivity	-85 dBm
	Energy detection	-85 dBm
	SNIR threshold	4 dB
Medium	Obstacle's loss type	“”
	Path loss type	FreeSpacePathLoss
	Path loss alpha	2
	Background noise type	IsotropicScalarbackgoundNoise
	Background noise power	-110 dBm
Scenario	Topology	{Bidirectional Highway, Manhattan grid}
	Maximum vehicle speed	33.33 m/s
	CAM message period	100 –1000 ms
	Simulation time limit	300 s
	Density of vehicles	{100-200-300-400-500}

Table 18 - Simulation parameter values

The parameter values used to model the individual nodes were selected based on those specified in the standards. A control channel (CCH) was used, with 10 MHz of bandwidth cantered at 5.9 GHz, channel number of 180 and default data rate of 6 Mbps. The nodes were configured to transmit with a power of 200mW or 23 dBm, which was below the 33 dBm power limit.

The road topologies were the bidirectional highway scenario and Manhattan grid scenario. Depending on the scenario the warm-up period and the simulation time differed. With the simulations the density of vehicles varies for low density (e.g., 100 vehicles) with high density (e.g., 500 vehicles).

6.1.4 Statistical Recording

In order to make statistics form the data recorded with the simulations, tools for data analysis are used: SQLite to manipulate the database, Python pandas to compute some values needed for the Histograms and Matplotlib to represent all the graphics correctly.

6.1.4.1 SQLite

SQLite is an in-process library that implements a self-contained, serverless, zero-configuration, transactional SQL database engine. SQLite reads and writes directly to ordinary disk files, and a complete database is stored in a single disk file.

OMNeT++ allows storing the data in a SQLite file disk format. With the data stored from the simulation the desired statistics could not be made for need of some parameters, for this reason, we use SQLite. With SQLite we can query the desired data from the simulation and stored in a new table with all the needed parameters to compute the statistics.

	index	eventNumber	simtimeRaw	Sender	Receiver	Sender_PosX	Sender_PosY	Receiver_PosX	Receiver_PosY	Distance	Error
	Filter	Filter	Filter	Filter	Filter	Filter	Filter	Filter	Filter	Filter	Filter
1	0	155	1.102359270484	22	11	529.044955...	598.4	440.992413...	598.4	88.0525426...	0
2	1	160	1.102359488887	22	35	529.044958...	598.4	401.6	522.862465...	148.149035...	0
3	2	165	1.102359586162	22	31	529.044959...	598.4	601.6	417.178584...	195.206135...	0
4	3	170	1.102359656886	22	39	529.044960...	598.4	594.169701...	401.840053...	207.067728...	0
5	4	175	1.102359662838	22	0	529.044960...	598.4	488.469267...	401.6	200.939361...	0
6	5	180	1.102359857406	22	44	529.044963...	598.4	398.4	387.443989...	248.134126...	0
7	6	185	1.102359948168	22	2	529.044964...	598.4	498.4	323.336840...	276.764982...	0
8	7	190	1.102360023352	22	16	529.044965...	598.4	401.6	309.424538...	315.830708...	0
9	8	195	1.102360100398	22	19	529.044966...	598.4	376.932816...	298.4	336.360083...	0
10	9	200	1.102360130362	22	1	529.044966...	598.4	201.6	521.085887...	336.448626...	0
11	10	205	1.102360177906	22	30	529.044967...	598.4	401.6	252.951506...	368.207660...	0
12	11	210	1.102360290363	22	9	529.044968...	598.4	498.4	220.372567...	379.267523...	0
13	12	215	1.102360311184	22	25	529.044968...	598.4	589.360939...	201.6	401.358015...	0
14	13	220	1.10236031408	22	4	529.044968...	598.4	529.703798...	198.4	400.000542...	0
15	14	225	1.102360390211	22	3	529.044969...	598.4	198.4	362.725606...	406.040042...	0
16	15	230	1.102360453178	22	27	529.044970...	598.4	132.741390...	398.4	443.910494...	0
17	16	235	1.102360552313	22	21	529.044971...	598.4	98.4	444.846786...	457.202013...	0
18	17	240	1.102360578972	22	8	529.044972...	598.4	281.109455...	201.6	467.891291...	0
19	18	245	1.102360652529	22	41	529.044972...	598.4	565.846828...	98.4	501.352547...	0
20	19	250	1.102360662517	22	20	529.044973...	598.4	101.6	336.820438...	501.131790...	0
21	20	255	1.102360688128	22	6	529.044973...	598.4	419.631900...	98.4	511.831242...	0
22	21	260	1.102360718487	22	43	529.044973...	598.4	39.8477672...	501.6	498.682410...	0
23	22	265	1.102360723117	22	24	529.044973...	598.4	385.694639...	101.6	517.068233...	0

Figure 18 - Packet error rate table with SQLite

For example, as we can see in Figure 18, we can get a table to compute the Packet Error Rate in function of distance.

6.1.4.2 Pandas

Pandas is an open-source Python package that is most widely used for data science/data analysis and machine learning tasks. It is built on top of another package named Numpy, which provides support for multi-dimensional arrays. As one of the most popular data wrangling packages, Pandas works well with many other data science modules inside the Python ecosystem, and is typically included in every Python distribution.

Moreover, Pandas makes it simple to do many of the time consuming, repetitive tasks associated with working with data, including: data cleaning, data fill, data normalization, Merges and joins, Data visualizations, Statistical analysis, Data inspection, etc.

Pandas is a useful tool for our project, it enables us to fill the tables of the databases with the missing data, and to compute the desired statistics in an easy and efficient way, and at the same time it allows us to represent and visualize the desired statistics. Moreover, Pandas supports the integration with many file formats or data sources, and one of them is SQL, which one we use for our project.

For plotting the data Pandas uses the power of Matplotlib, a powerful python library for creating static, animated and interactive visualization. Matplotlib is a cross-platform, data visualization and graphical plotting library for python and its numerical extension Numpy. It offers a viable open-source alternative to MATLAB.

With the usage of these tools, we are able to analyse different simulations scenarios and extract statistical performance.

6.2 Robotec V2X Simulation (RTC)

Robotec.ai is developing a V2X Simulator as ROS2 module, that can be integrated with any AV simulator having support for ROS communication. The simulator works as external module with replicated entire simulated environment, both static scene and all dynamic objects (traffic agents). In the CPSoSaware project, Robotec V2X Simulator will be used together with Robotec Real World Simulator working as AV simulation, mostly to validate cooperative awareness scenarios in the automotive pillar of the project.

6.2.1 Simulation Framework Overview

The main component of V2X Simulator is the Environment that represents the simulated world. Each of the communicating objects sends and receives V2X messages that are propagated in space using representation expressed by selected Propagation Model. Eventually, several propagation models with different levels of realism will be developed and supported in Robotec V2X Simulator.

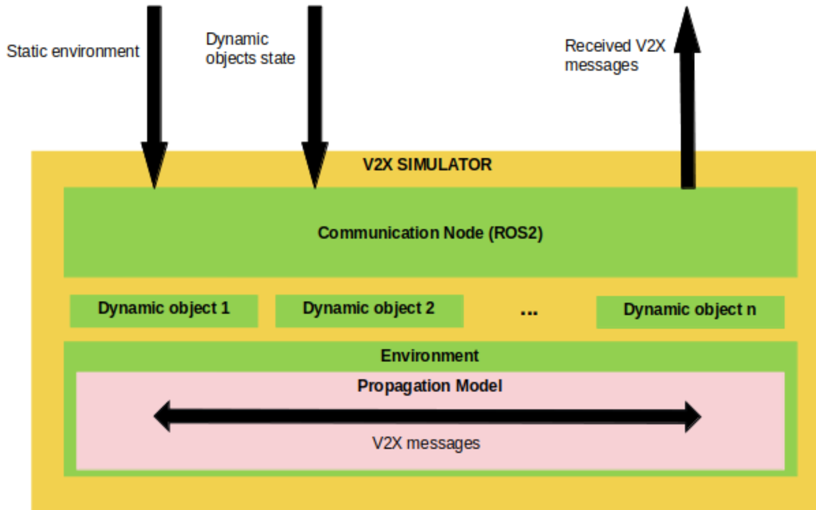


Figure 19 – V2X Simulator structure

Integration with AV simulator consists of 3 types of interfaces:

- Static environment communication – ROS message containing information about the scene. Environment is created in V2X simulation only once, on initialization of a simulated use case.
- Dynamic objects state – ROS messages sent periodically from each traffic agent. This message is responsible for sharing locations of agents and all the data transmitted in V2X message from AV simulator to V2X Simulator.
- Received V2X messages – ROS message sending all received V2X messages back to AV Simulator

The data is supplied from AV Simulator to the V2X simulator, then inside the V2X module the network propagation is modelled and all the messages that reached target receivers are sent back to the AV simulator. Information received from other traffic agents can be further processed in AV Simulator and used as extension to perception algorithms and can improve safety or help in more optimal path planning.

6.2.2 Simulation scenarios

Robotec V2X simulator will be used for validation of cooperative awareness use cases of the automotive pillar of the project. Unlike the OMNET++/SUMO simulator, Robotec V2X simulator will not be used for analysis of Inter communication itself, but to assess the benefit of cooperative awareness in selected scenarios from the automotive domain.

The use case is defined as a set of scenarios defined by PASEU that will be tested in the simulated environment. Specific scenario will be described using JSON configuration file with following structure of parameters:

- Application name - name of simulation application used in the test case.
- Agents - list of all traffic agents used in the test case.
- Name – name of the traffic agent
- Type – type of the traffic agent (car/truck/pedestrian)

- Path - name of the path on which the traffic agent is located.
- Point – point id on the path describing starting position of the traffic agent.
- Solver – used solver controlling movement of the traffic agent.
- Speed – desired speed of the traffic agent (in m/s)

Example of configuration file for scenario with 2 cars and 2 pedestrians is presented below:


```
{
    "application_name":"UrbanApp",
    "agents": [
        {"name": "Car1", "type": "car", "point": 1, "path":"Path_right.obj", "solver": "pathFollower", "speed": 7.0},
        {"name": "Car2", "type": "car", "point": 1, "path":"Path_left.obj", "solver": "pathFollower", "speed": 7.0},
        {"name": "Pedestrian1", "type": "pedestrian", "point": 1, "path":"Path_crossing.obj", "solver": "pathFollower",
                "speed": 1.5},
        {"name":"Pedestrian2","type":"pedestrian","point":20,"path":"Path_crossing.obj","solver":"pathFollower",
            "speed": 1.5}
    ]
}
```


With such format of configuration files, multiple scenarios can be easily created and validated in the simulation environment. To properly assess the benefit of introducing cooperative situational awareness, meaningful metrics should be calculated for all described scenarios. Comparison of metrics values for cases with and without inter communication enabled will produce numerical results of cooperative awareness algorithms developed in CPSoSaware project. Detailed metrics used for validation of this scenario will be defined in Definition and planning of Evaluation Trials (D6.4).

6.3 Results

This section presents the results for the experimental evaluation of Cooperative Awareness basic service using the modified IEEE 802.11p simulator. Its performance was studied by varying different parameters such as vehicle density, packet length. The plots were made according to the Tx-Rx distance, defined as the distance between two communicating nodes.

6.3.1 Packet Error Rate

Packet Error Rate (PER) is used to test performance of an access terminal's receiver. PER is the ratio, in percent, of the number of packets not successfully received by the access terminal. The resulting PERs of the different scenarios were used to better understand the behaviour. In this project the PER was defined as follows.

$$PER = 100 \cdot \frac{Erroneous\ packets}{Total\ received\ packets\ above\ SNIR\ threshold}$$

A CAM is considered to be correctly received when it reaches the MAC layer without errors, in addition to satisfying the receiver sensitivity, energy detection and SNIR threshold values in the physical layer.

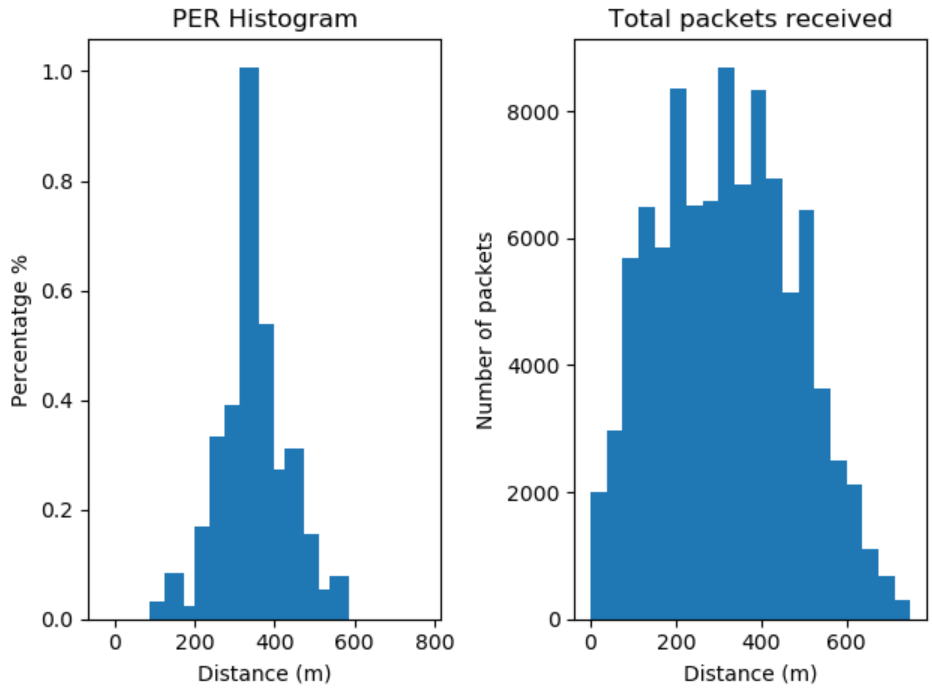


Figure 20 - PER Histogram for vehicle density of 50

7 Conclusion

This deliverable has defined the Inter and Intra Communication Models in the CPSoSaware project, and the set of tools and simulators used for them. Also, the supporting developments to gather and process the information have been shown, linking the work with WP4 and WP6 where the models will be used to simulate and implement different use cases respectively.

8 References

[1] European Telecommunications Standards Institute (ETSI). “Intelligent Transport Systems (ITS); Communications Architecture” ETSI EN 302 665 V1.1.1 (2010-09).

[2] European Telecommunications Standards Institute (ETSI). “Intelligent Transport Systems (ITS); Vehicular Communications; Basic Set of Applications; Part 1: Functional Requirements” ETSI TS 102 637-1 V1.1.1 (2010-09).

[3] European Telecommunications Standards Institute (ETSI). “Intelligent Transport Systems (ITS); Vehicular Communications; GeoNetworking; Part 5: Transport Protocols; Sub-part 1: Basic Transport Protocol” ETSI EN 302 636-5-1 V1.2.1 (2014-08).

[4] European Telecommunications Standards Institute (ETSI). “Intelligent Transport Systems (ITS); Vehicular Communications; GeoNetworking; Part 1: Requirements” ETSI EN 302 636-1 V1.2.1 (2014-04).

[5] European Telecommunications Standards Institute (ETSI). “Intelligent Transport Systems (ITS); Access layer specification for Intelligent Transportation Systems operating in the 5 GHz frequency band” ETSI EN 302 663 V1.2.1 (2013-05).

[6] T. S. Rappaport, Wireless Communications.: Principles and Practice.Prentice Hall, 1996.

[7] M. Stoffers and G. Riley, "Comparing the ns-3 Propagation Models," 2012 IEEE 20th International Symposium on Modeling, Analysis and Simulation of Computer and Telecommunication Systems, 2012, pp. 61-67, doi: 10.1109/MASCOTS.2012.17.

[8] "IEEE Standard for Low-Rate Wireless Networks," in IEEE Std 802.15.4-2015 (Revision of IEEE Std 802.15.4-2011) , vol., no., pp.1-709, 22 April 2016, doi: 10.1109/IEEESTD.2016.7460875.

[9] SUMO - Wiki. [Online]. Available: https://sumo.dlr.de/docs/ [Accessed: April 20, 2021]

[10] SQLite - Wiki. [Online] Available: https://www.sqlite.org/about.html [Accessed: April 30, 2021].

[11] Pandas – Wiki. [Online]. Available: https://pandas.pydata.org/docs/index.html [Accessed: April 30 ,2021]

[12] Sammour, I.; Chalhoub, G. Evaluation of Rate Adaptation Algorithms in IEEE 802.11 Networks. Electronics 2020, 9, 1436. https://doi.org/10.3390/electronics9091436