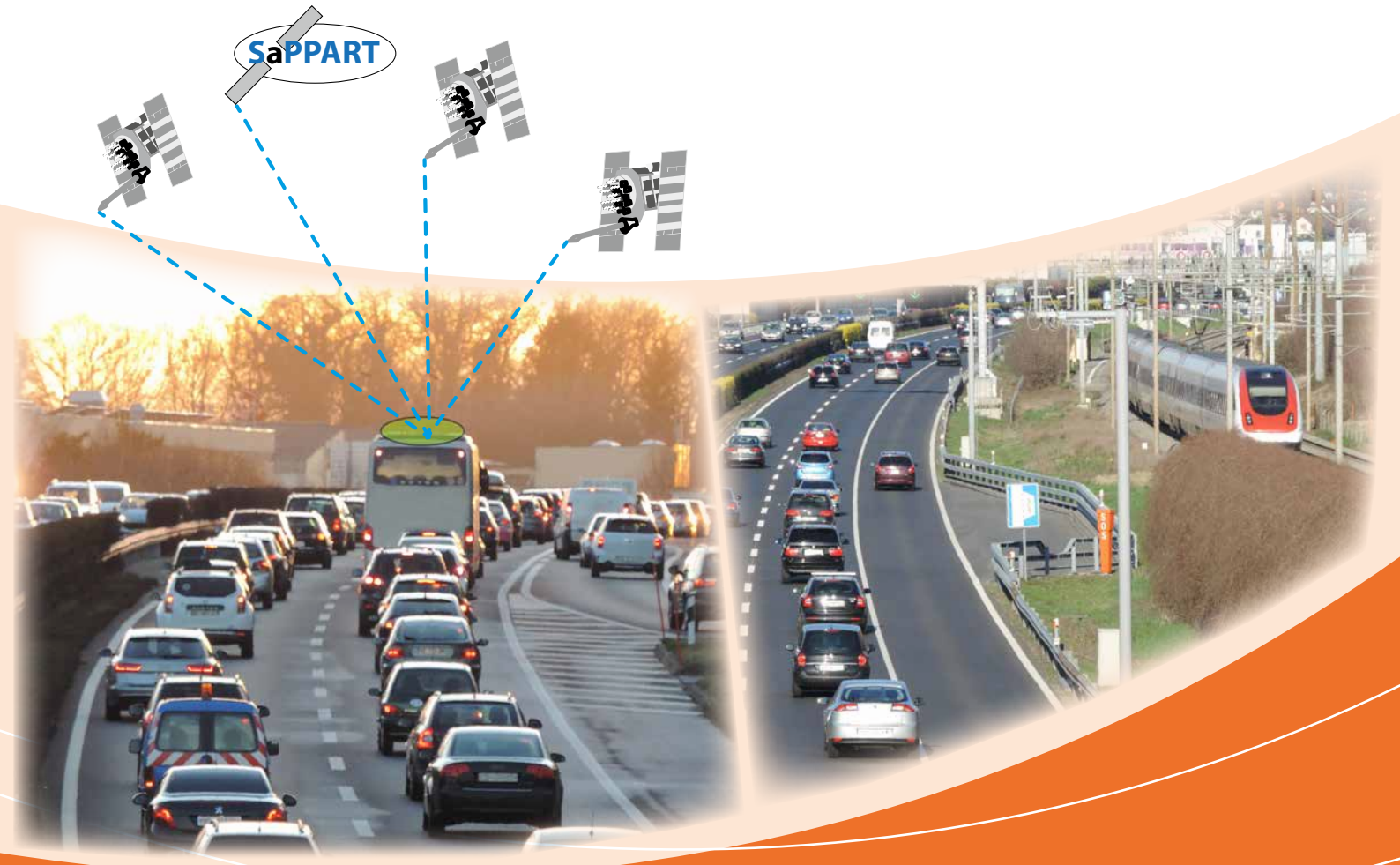
TECHNIQUES ET MÉTHODES

SaPPART Handbook

Assessment of positioning performance in ITS applications



Satellite Positioning Performance Assessment for Road Transport
COST Action TU1302

TMI 2

SaPPART Handbook

Assessment of positioning performance in ITS applications

June 2017

TECHNIQUES ET MÉTHODES

This handbook is the second deliverable of SaPPART COST Action. The first deliverable is the white paper and the last publication of the Action will be the guidelines.
It was written under the direction of François Peyret and Pierre-Yves Gilliéron, respectively chair and vice-chair of the Action.

The contributors are (by alphabetical order):
David Bétaille, Institut français des sciences et technologie des transports, de l'aménagement et des réseaux (Ifsttar), France
Jesper Engdahl, Rapp Trans SA, Switzerland
Vasillis Gikas, National Technical University of Athens, Greece
Pierre-Yves Gilliéron, École polytechnique fédérale de Lausanne (EPFL), Switzerland
Shaojun Feng, Imperial College, United Kingdom
Ola Martin Lykkja, Q-Free, Norway
Mihai Niculescu, Intelligent Transport Systems Romania, Romania
Miguel Ortiz, Institut français des sciences et technologie des transports, de l'aménagement et des réseaux (Ifsttar), France
Washington Ochieng, Imperial College, United Kingdom
Harris Perakis, National Technical University of Athens, Greece
François Peyret, Institut français des sciences et technologie des transports, de l'aménagement et des réseaux (Ifsttar), France
Laura Ruotsalainen, Finnish Geospatial Research Institute, National Land Survey (FGI NLS), Finland
Marko Ševrović, University of Zagreb, Croatia
Ioanna Spyropoulou, National Technical University of Athens, Greece
Rafael Toledo Moreo, Universidad Politécnica de Cartagena, Spain
Silvia Andreea Voicu, Intelligent Transport Systems Romania, Romania

This publication is supported by COST. Neither the COST Association nor any person acting on its behalf is responsible for the use which might be made of the information contained in this publication. The COST Association is not responsible for the external websites referred to in this publication.

Please refer to this work as follows:

COST Action TU1302. SaPPART Handbook: Assessment of positioning performance in ITS applications. Ifsttar, 2017. techniques et méthodes, TMI 2. 77p. ISBN 978-2-85782-727-6

French Institute of Science and Technology for Transport, Development and Networks - Ifsttar
14-20 boulevard Newton - Cité Descartes - Champs-sur-Marne - 77447 Marne-la-Vallée cedex 2
www.ifsttar.fr
Les collections de l'Ifsttar – techniques et méthodes – TMI 2
ISBN 978-2-85782-728-3– ISSN: 2492-5438

Table of contents

Chapter 3.
Sensitivity analysis: matching positioning performance to the required E2E performance

Executive Summary

The way intelligent road transport systems make use of positioning information is very variable. In the majority of systems, positioning information is processed by complex algorithms such as map-matching and geofencing, which can compensate for shortcomings in the positioning information and maintain acceptable performance. For most Intelligent Transport Systems (ITS), the impact of the quality of the positioning information on ITS user service level performance cannot be easily estimated. However, it can be of fundamental importance for critical services, and therefore requires detailed analysis.

Since the behaviour of positioning systems, especially those based on the Global Navigation Satellite System (GNSS), is highly dependent on the operational scenario, this *Handbook* proposes an approach based on a method called *Sensitivity analysis*. This is carried out with positioning data obtained either under real conditions (field data capture campaigns) or simulated using realistic validated error models. An example of such an error model and how it has been specified is presented in Appendix A of this document.

The *Sensitivity analysis* proposed in this *Handbook* is a general method that makes it possible to match the performance of the positioning terminal with the end-to-end (E2E) performance of the ITS system. It measures how sensitive the Key Performance Indicators (KPIs) of the system are to the quality of the position. It is based upon the generation of a high number of simulated tests, applying trajectories that are degraded using a realistic positioning error model derived from real data. These real data are obtained through field tests that are representative of the operational scenarios. Two examples are used to demonstrate the applicability and benefits of the proposed *Sensitivity analysis* method.

Moreover, prior to the *Sensitivity analysis*, the key performance features of the positioning terminal and their associated metrics must be rigorously defined. The basic key features are availability, accuracy, integrity, time-to-first-fix (TTFF), but some other timing performance features also need to be considered. The metrics are built on errors with respect to a reference which are realizations of random variables and therefore should be based upon the statistics of experimental data. This document proposes to use the 50th, 75th and 95th percentiles of the Cumulative Distribution Function (CDF) of the errors. On this basis, it is straightforward to define three performance classes based upon two sets of numerical values of these percentiles.

The setting up of a certification reference framework is necessary to boost the development of GNSS-based applications and absolutely vital for safety-critical application, such as autonomous driving. This framework should define commonly agreed performance classes and test procedures, ideally in the form of a standard.

Introduction

This document is the second deliverable of the COST Action SaPPART, a European network of scientists and stakeholders that aims to promote smart use of GNSS technology in the field of road transport and intelligent urban mobility.

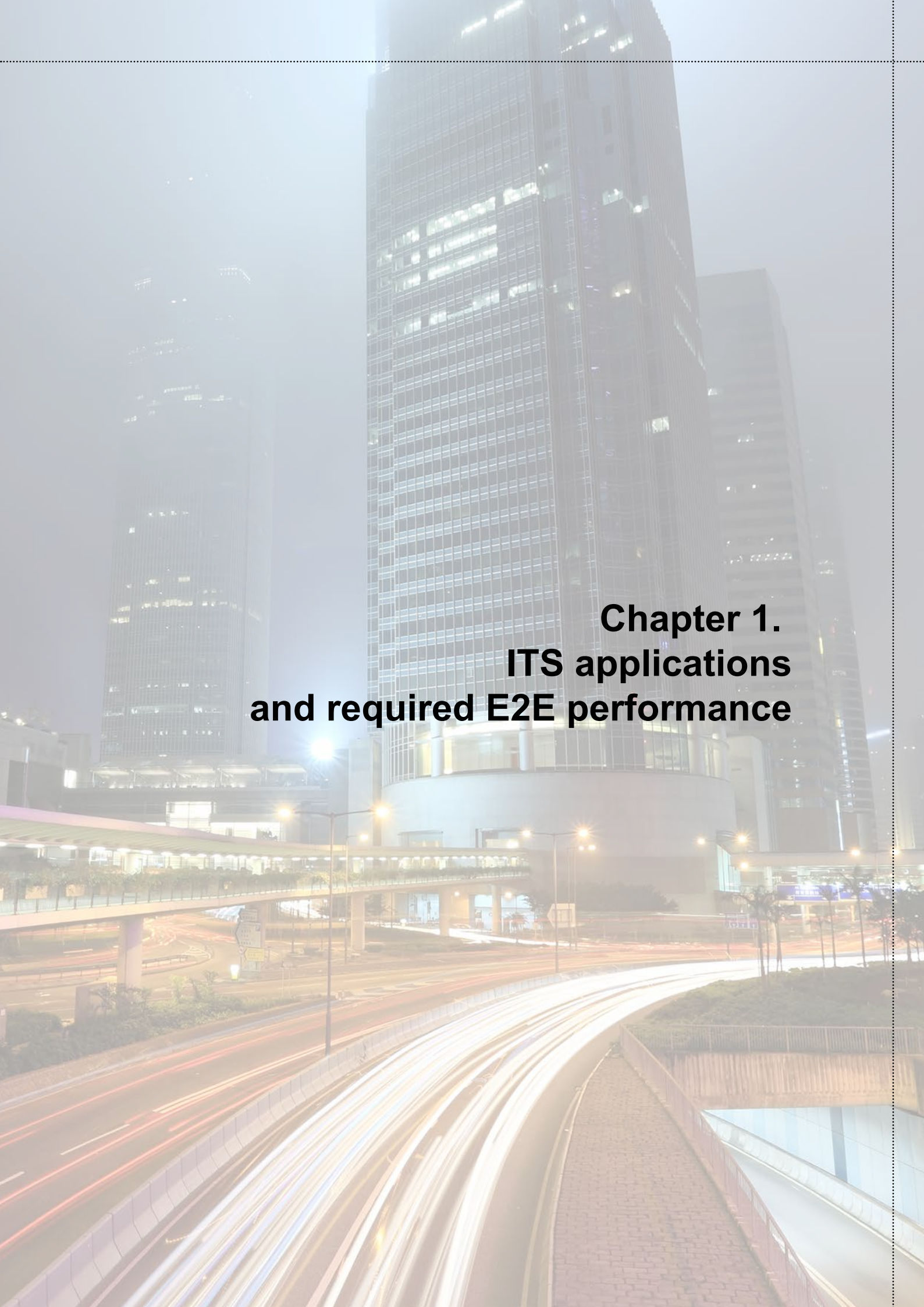
The first deliverable, called the “White Paper” [1], introduced the fundamentals of positioning systems, with the focus on GNSS, explaining their role in transportation and stressing the importance of a reliable assessment of their performance.

In this document, which has the title “Handbook - Assessment of positioning performance in ITS applications”, the major issue of performance assessment is described and presented in an informative manner.

Chapter 1 recalls the non-straightforward nature of the use of positioning information in some representative road transport systems. Specifically, four applications have been chosen because of their growing importance in the ITS sector and the different ways in which they process the raw positioning information to deliver their final services to users. The conclusion of this chapter is that it is in general not possible to estimate the level of performance of the positioning terminal required to deliver the ITS user service level performance without a thorough analysis of the sensitivity of the ITS service to the quality of positioning.

Chapter 2 discusses positioning performance at the terminal level. In particular, this chapter focuses on Position, Velocity and Time (PVT) errors and introduces a parametric error model that has been developed in the framework of SaPPART and is presented in Appendix A. Such a model can be very useful when a large number of tests, representing different operational conditions, have to be carried out to characterize comprehensively the positioning performance of a given terminal. Chapter 2 ends by presenting the positioning metrics recommended by the standardization group on “Navigation and positioning receivers for road applications” (i.e. CEN/CLC/TC 5/WG1).

Chapter 3 provides simple informative examples of the *Sensitivity analysis* method introduced in Chapter 1 by applying it to two ITS services, namely Road User Charging and eCall. These examples also illustrate how sensitive E2E performance is to PVT performance.

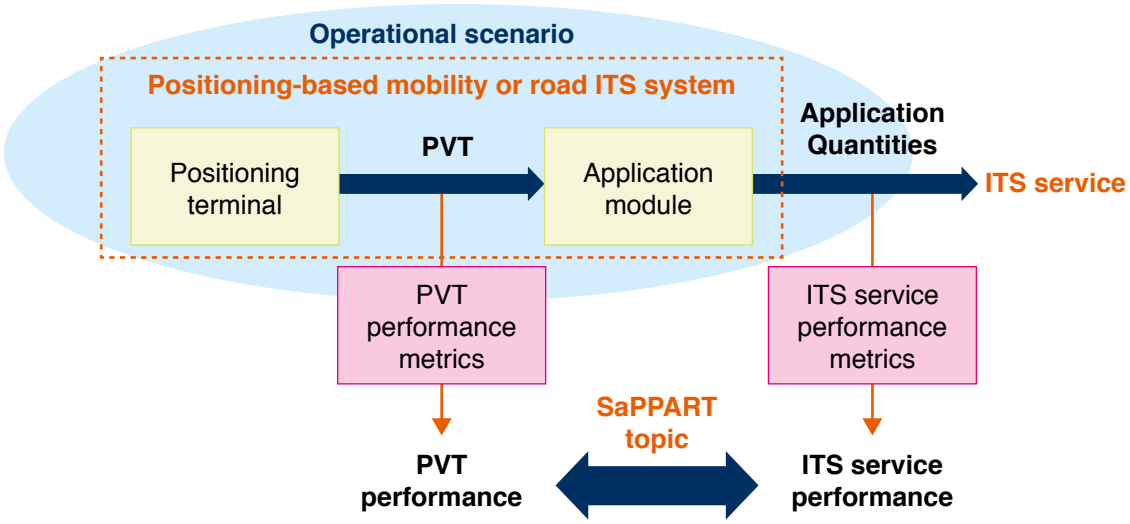


Chapter 1.
ITS applications and required E2E performance

Any intelligent transport system relying on positioning can be broken down into two main sub-systems: the positioning terminal and the application module, as illustrated in figure 1 below.

In SaPPART, the term “application” (or “application module”) refers to the software component that processes the PVT data (with other application specific data) in order to derive or determine the “application quantities” on which the final ITS service is built.

Figure 1
Fundamental scientific issues covered by SaPPART



SaPPART investigates the relationship between the performance of the GNSS-based positioning terminal (GBPT) at the PVT level and the performance of the whole system at the service level. This relationship can be analysed in both directions, namely:

- From the positioning terminal to the application module. This consists of assessing whether or not the positioning performance fulfils the final ITS service requirements.
- From the application module to the positioning terminal. This consists of deriving the necessary minimum performance of the positioning terminal at the PVT level from the E2E or ITS service level performance requirements.

In this chapter, some examples of major location-based ITS services are analysed with respect to these relationships. Since there are a large number of location-based ITS services, examples representing four of the main relevant families are used here. These are Road User Charging (RUC), Emergency Call (eCall), Intelligent Speed Adaptation (ISA) and one example of an Advanced Driver Assistance System (ADAS).

The features of each of the specific examples that is representative of the families are described and analysed taking into account:

- the precise use case and the operational scenario that is considered
- the E2E performance indicators (or metrics) and targets of the complete system, which are directly dependent on the position information
- the information needed by the application
- the way this information is processed to deliver the final service
- the nature of the relationship between E2E performance and PVT performance

The following sections present the analyses of the four selected applications. These elements are re-used in Chapter 3 to provide examples of the Sensitivity analyses for the first two applications.

1.1 Application analysis

1.1.1 RUC

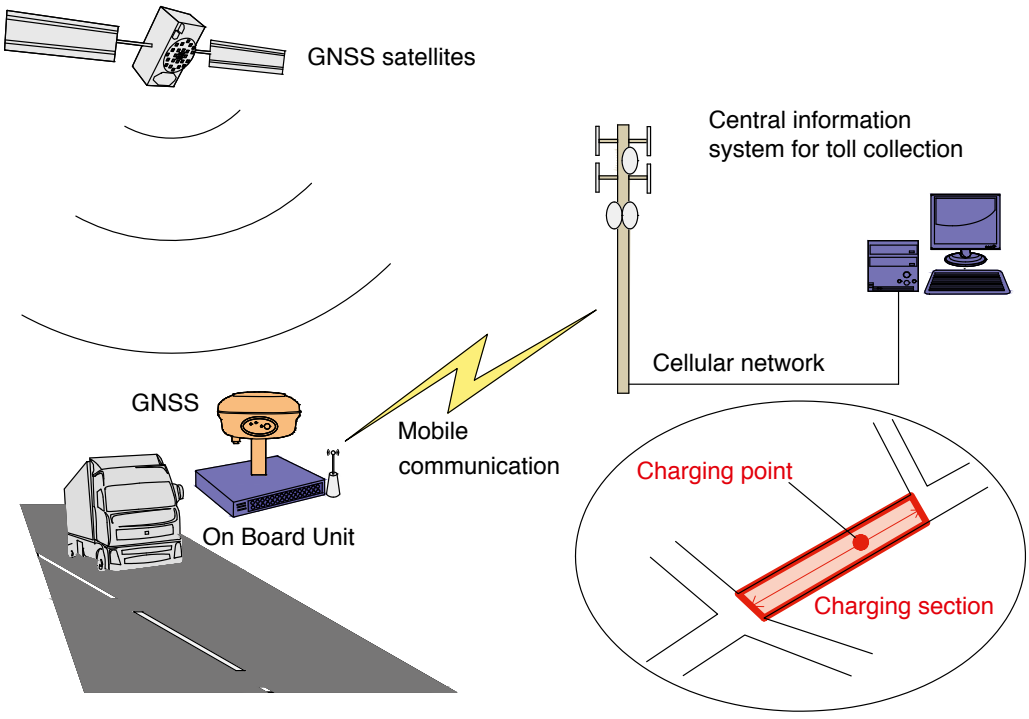
Road User Charging (also known as Road Tolling, Road Charging and Congestion Charging, for example) consists of charging road users for driving on specific roads or infrastructures. Variations include:

- Event-based charging – e.g. tunnels and bridges
- Area-based (or Cordon-based) charging – paying to drive in a particular area (on crossing a cordon), irrespective of the distance driven, e.g. the London, Stockholm or Oslo congestion charging schemes
- Distance-based charging – paying an amount that is proportional to the distance driven. There are a number of sub-variants such as road segment charging, pure distance-based charging (often called kilometre charging) and others. According to ISO/TS 17444-1, it is crucial to distinguish between “continuous charging” where charges are based on measuring the distance that is actually travelled and “discrete charging” where charges are based on the length of the travelled road segments. This is especially critical when assessing the performance of an application.

All the above variants may in turn be varied on the basis of time, vehicle class, road type, direction of travel, current level of congestion, etc.

Figure 2 illustrates the principle of a GNSS-based RUC system.

Figure 2
Principle of a GNSS-based RUC system



The specific use case corresponding to distance-based charging, more precisely segment-based (“discrete charging”), is analysed in the table below.

Table 1
RUC use case & operational scenario

Use case & operational scenario	E2E performance indicators (indicative targets)	Information needed by the application, from GBPT or other sources	Application data processing	Relationship between E2E and PVT performances
Road segment based national highway charging, proportional to distance driven. The distance is computed by summing the conventional lengths of the charging sections Different kinds of environments, from countryside to peri-urban, but not deep urban	E2E overcharging rate (for a travelled segment) ($< 10^{-6}$) E2E undercharging rate (for a travelled segment) (< 3%)	From the positioning terminal: - position, time (mandatory) - velocity (optional) - integrity quantities (optional) From other sources: - location of charging points, - map of road network	Geofencing on a geo-object built around the charging point Map-matching (optional)	Very complex Depends on: - the location and configuration of the charging points - the traffic on the charging points - the traffic in the neighbourhood of the charging points - the performance of the geofencing algorithm - map quality

Note on the geofencing algorithm

The geofencing algorithm is, along with the positioning terminal, very much a key element of an RUC system. The accurate and reliable performance of the geofencing algorithm can compensate for the poor performance of the positioning terminal.

There are many geofencing algorithms for detecting charging events at a charging point. A simplified scheme is shown below as one example among many others.

Figure 3
Example of a simplified geofencing scheme

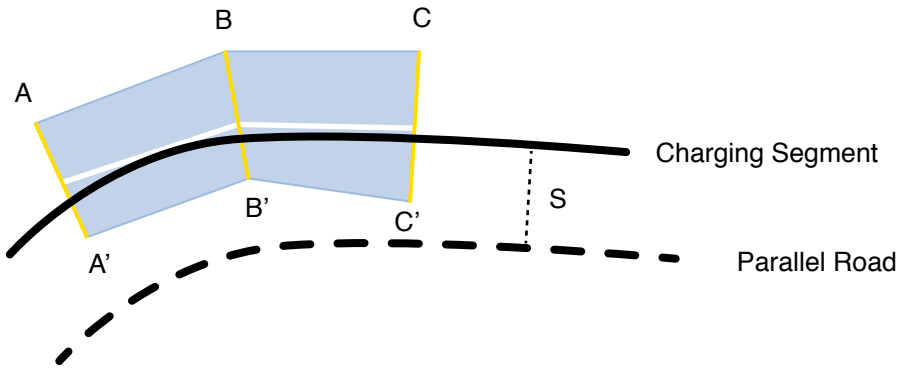


Figure 3 shows a “geo-object”, also called a “virtual gantry” (the grey area on the figure), that is virtually placed on the charging segment. If a vehicle is detected as having crossed the three transversal segments AA’, BB’ and CC’, it is considered to have passed through the charging segment.

The geo-object must be designed in an appropriate manner, taking into account the configuration of the charging segment as well as the other non-chargeable roads in the neighbourhood. Its length will depend on the geometry of the charging segment while its width, which is much more critical, will depend on the distance (denoted by S in figure 3) to the closest non-chargeable parallel road. If S is small, in the case of a dense network, the geo-object should not intersect the parallel road and the transversal segments must be small to prevent false charging, with the risk of missing some vehicles if the positioning accuracy is too poor.

1.1.2 eCall

The eCall service allows a user to initiate an automatic 112 emergency call in the event of a serious accident. The alarm is triggered on the basis of information from on-board vehicle sensors such as airbags, seatbelt tensions, or deceleration sensors. The call can also be generated manually by pushing a button. When a call is generated, a digital Minimum Set of Data (MSD) including the position of the damaged car is sent first after which a voice communication channel opens. The MSD is received and processed by Public-Safety Answering Points (PSAPs) where operators perform map-matching to determine the physical location to which emergency teams are dispatched. The eCall system described here is the one defined and endorsed by the EU Regulation [9] and European standards [10-12]. Note however, that before eCall there were, and still are, systems with similar functionality, provided by private car manufacturers.

The specific use case analysed below corresponds to the Pan-European system as defined in the European standards.

Table 2
eCall use case & operational scenario

Use case & operational scenario	E2E performance indicators and (indicative performance targets)	Information needed by the application, from GBPT or other sources	Data processing applied by the application	Nature of the relationship between E2E performance and PVT performance
Pan-European eCall generates an automatic 112 emergency call and transmits an MSD containing the absolute position of the damaged vehicle At the PSAP, map-matching is performed to determine the physical location of the accident to which the emergency vehicles are dispatched All roads, all environments	Probability of correct detection of the road (> 99%) Probability of correct detection of the carriageway (> 95%) Accuracy (95th percentile) of the longitudinal position along the road (< 50 m)	From the positioning terminal: - position at the triggering moment (mandatory) - vehicle direction, or heading (mandatory) - previous 3 positions, with no time tag for privacy reasons (optional) - confidence bit (optional) - speed not allowed for privacy reasons From other sources: - map of the road network	Map-matching (at the carriageway level) performed off-line at the PSAP	Apparently quite simple from a pure positioning point of view Can be depicted by simple off-line map-matching, but the difficulty comes from the poor PVT information allowed by the standard (for privacy reasons)

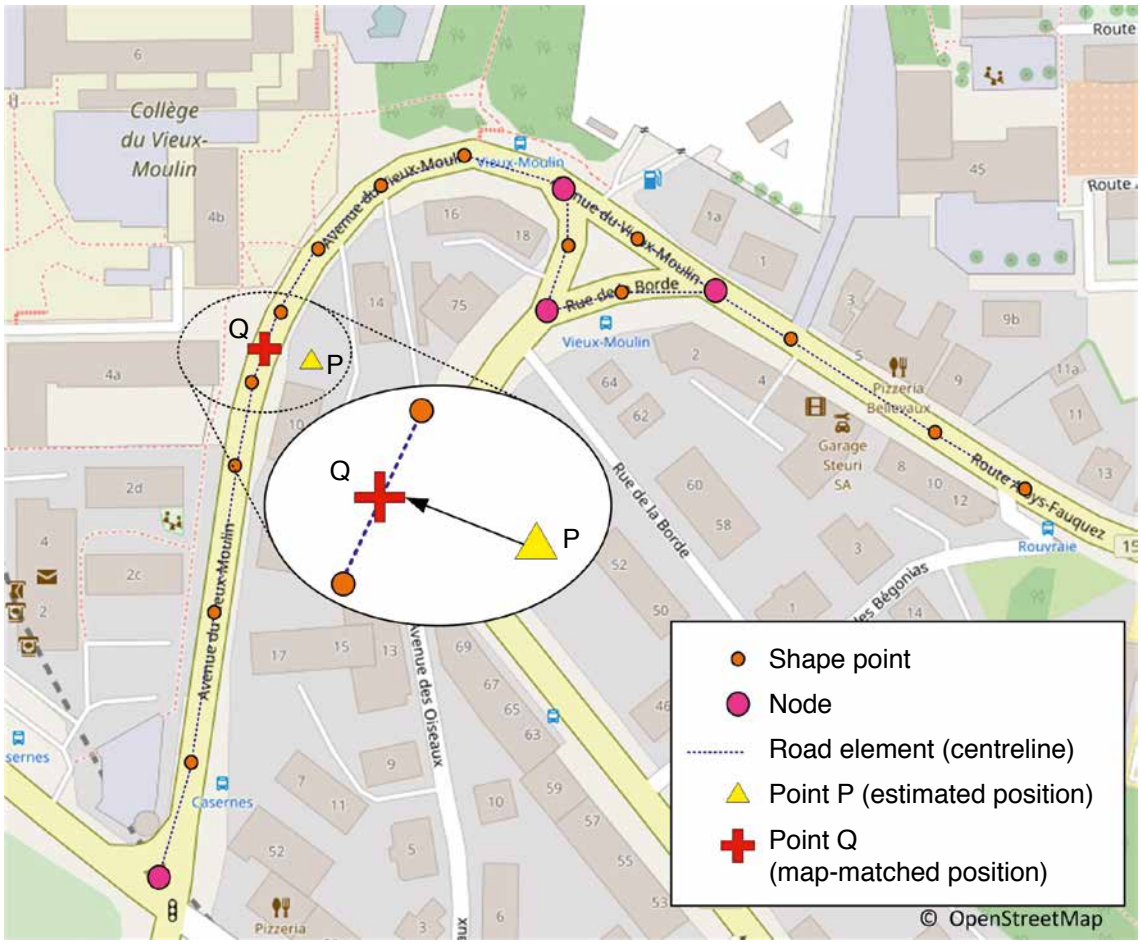
Note on the map-matching process

Map-matching consists in transferring a 2D position expressed in an absolute reference frame (e.g., in terms of longitude and latitude when using a geodetic reference frame such as WGS 84), to a local reference frame on a digital map. This means that the position parameters (the coordinates) will no longer be the longitude and latitude, but parameters allowing the vehicle to be localized on a road map, such the node number, direction, and distance to the node.

These parameters depend on the way the road is modelled on the map. A road may consist of a single carriageway (with 2 or 3 lanes) but it can also be a dual carriageway with, generally, one carriageway for each driving direction. In Europe dual carriageways generally have two or three lanes (highways or motorways). Single carriageway roads are usually modelled in digital maps by a polyline representing the centreline of the carriageway. A polyline is a series of line segments joining points that can be either *Nodes* (located at intersections) or *Shape points* (artificially added in curves). These points are geo-referenced, meaning that their coordinates are known in an absolute reference frame. Dual carriageway roads are generally modelled by using two polylines.

In more sophisticated maps, the polyline can be replaced by a spline or a clothoid. A clothoid (also called Euler spiral or Cornu spiral) is the exact geometrical figure that is used to design roads and it can be easily broken down into an arc of a circle or a straight line.

Figure 4
Principle of a digital road map and of the map-matching process



Performing map-matching consists of “projecting” point P, whose absolute coordinates are known, onto point Q on the map (figure 4). The parameters that localize point Q on the map are generally the identification (ID) value of the segment and the distance of point Q from the origin node of the segment. As road models differ from one map manufacturer to another, this set of parameters is manufacturer-dependant.

Many different map-matching methods exist in reality, ranging from the basic orthogonal projection onto the nearest road segment, to much more sophisticated methods taking into account other parameters such as past locations, the direction or the dynamic behaviour of the vehicle.

Lanes are generally not modelled, in which case it is impossible to know, from the map-matched position, where the vehicle is in the cross-lane direction. Nevertheless, some research has been done using advanced digital maps that describe all the lanes on the road and therefore allow accurate map-matching at lane level.

1.1.3 ISA

Intelligent Speed Adaptation is also known as Intelligent Speed Limiter, Intelligent Speed Assistance, Speed Alerting or Intelligent Speed Authority.

The purpose of ISA is to mitigate speeding, i.e. drivers driving at speeds above the legal speed limit. This is accomplished by informing or alerting the driver, and even controlling the vehicle, depending on the system design. Depending on the Human Machine Interface (HMI), the warnings can be visual or audio or use haptic/tactile feedback. The latter may involve vibration or increased upward pressure on the accelerator pedal or even control of the vehicle, depending on the variant of the system.

Three different operating modes can be used:

- Informative ISA: when the system provides information on the speed limit throughout the ISA-deployed zone
- Warning ISA: when the system provides a warning to the driver but does not control vehicle speed
- Intervening ISA: when the system controls vehicle speed. This can either be voluntary when the system can be overridden or compulsory when the system cannot be overridden.

The prevailing speed limit is generally obtained by a real-time map-matching process that requires localisation via GNSS and a digital map with up-to-date speed limit information. The system may also cater for variable speed limits provided that they are available. For instance, the speed limit could be varied according to the prevailing traffic conditions, the weather, road alignment (e.g. road curvature) or a combination of these.

The use case analysed in the table below corresponds to the generic one, and makes no assumptions concerning the operating mode or the way the prevailing speed limit is obtained.

Table 3
ISA Use case & operational scenario

Use case & operational scenario	E2E performance indicators and (indicative performance targets)	Information needed by the application, from GBPT or other sources	Data processing applied by the application	Nature of the relationship between E2E performance and PVT performance
The driving speed is monitored and compared with the prevailing speed limit on the road (or the lane) the vehicle is on	- Percentage of speed transitions correctly detected (< 99%) - Percentage of speed transitions falsely detected (<0.1%) - Percentage of speed transitions not detected (< 1%) within a defined duration or distance from a transition reference point (10 m)	From the positioning terminal: - 2D position - Speed (optional) - Time (if the system caters for variable speed limits) From other sources: - Digital map with speed limit information - For road segments where speed limits change over time, on-line information (i.e. traffic or weather information) coming through a communication channel) - Speed provided by other equipment (speedometer) rather than from the GBPT - Vehicle type (e.g., car, truck, with/without trailer…)	- Real-time longitudinal map-matching (or map-matching on-road segment) for curvilinear determination - Map-matching on-lane (for cases where speed limit varies between traffic lanes) - Driving speed extraction from GNSS data or the vehicle's speedometer - Time synchronisation (for variable speed limits)	Apparently quite simple from a pure positioning point of view Basically a real-time map-matching process, but can be challenging in the case of lane-level map-matching

Note on the lane-matching process

For certain cases, different speed limits can be applied to specific parts of the road. The speed limit may depend on the direction of travel or on the travelled lane, and may even vary on the same carriageway (e.g. a slow lane for heavy vehicles with a lower speed limit than the other lanes).

In these cases, a lane level map-matching (or lane-matching) process is mandatory, even though this is still very challenging with even today's state-of-the-art real-time positioning techniques. Moreover, lane-matching requires lane level digital maps, which are not widely available.

1.1.4 ADAS

Advanced Driver Assistance Systems are systems that have been developed to automate, adapt and enhance vehicle systems for safety and better driving. The safety features are designed to avoid collisions and accidents by means of technologies that warn the driver about potential problems, or to avoid collisions by implementing safeguards and taking over control of the vehicle. Adaptive features may automate lighting, provide adaptive cruise control, automate braking, incorporate traffic warnings, alert the driver to other cars or dangers, keep the driver in the correct lane, or show what is in blind spots. Many forms of ADAS are available. Some features are built into cars, others are offered as add-on packages (incl. aftermarket kits). ADAS are one of the fastest-growing segments in automotive electronics.

From among the huge variety of ADAS, we have selected the use case of the “Intersection Collision Risk Warning” [13] (ICRW) cooperative application as an example here.

Figure 5 below illustrates this application. The trajectories of vehicles are extrapolated from the current instant, as is an estimation of their pose (position and direction).

Figure 5
The ICRW use case

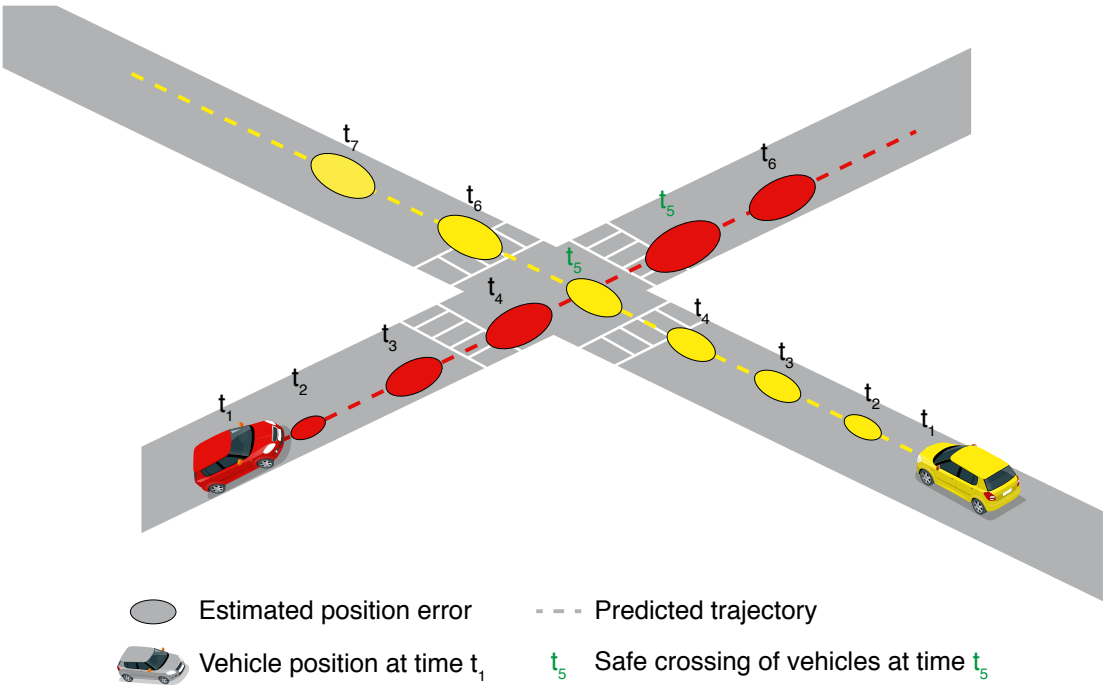


Table 4
ADAS use case & operational scenario

Use case & operational scenario	E2E performance indicators and (indicative performance targets)	Information needed by the application, from GBPT or other sources	Data processing applied by the application	Nature of the relationship between E2E performance and PVT performance
Cooperative Intersection Collision Risk Warning Driver warning in risk situation: - vs. another vehicle, - vs. other dynamic obstacles, - vs. static obstacles	Missed warning rate ($< 10^{-6}$) False warning rate ($<10^{-4}$)	From GBPT (probably hybridized): - position, - speed, - direction (heading) From other sources: - position, speed and heading of other nearby vehicles, - position of all the static obstacles	Proposition: extrapolation of the trajectories and their uncertainties from the current pose (assuming constant speed and curvature) and estimation of the risk of collision	Quite complex Depends not only on the quality (exactness) of the current position of the vehicle and the surrounding objects, but also on the quality of the derivatives of the trajectory(ies): velocity vector, curvature…

Note: it can be considered that a correct warning is a warning communicated to the driver when the estimated time-to-collision is less than a given safety threshold, e.g. 2 s. A false warning corresponds to a warning that is communicated to the driver when the

estimated time-to-collision is either higher than the safety threshold or not available. A missed warning occurs when the actual time-to-collision is below the threshold but the system fails to send a warning.

1.2 Summary

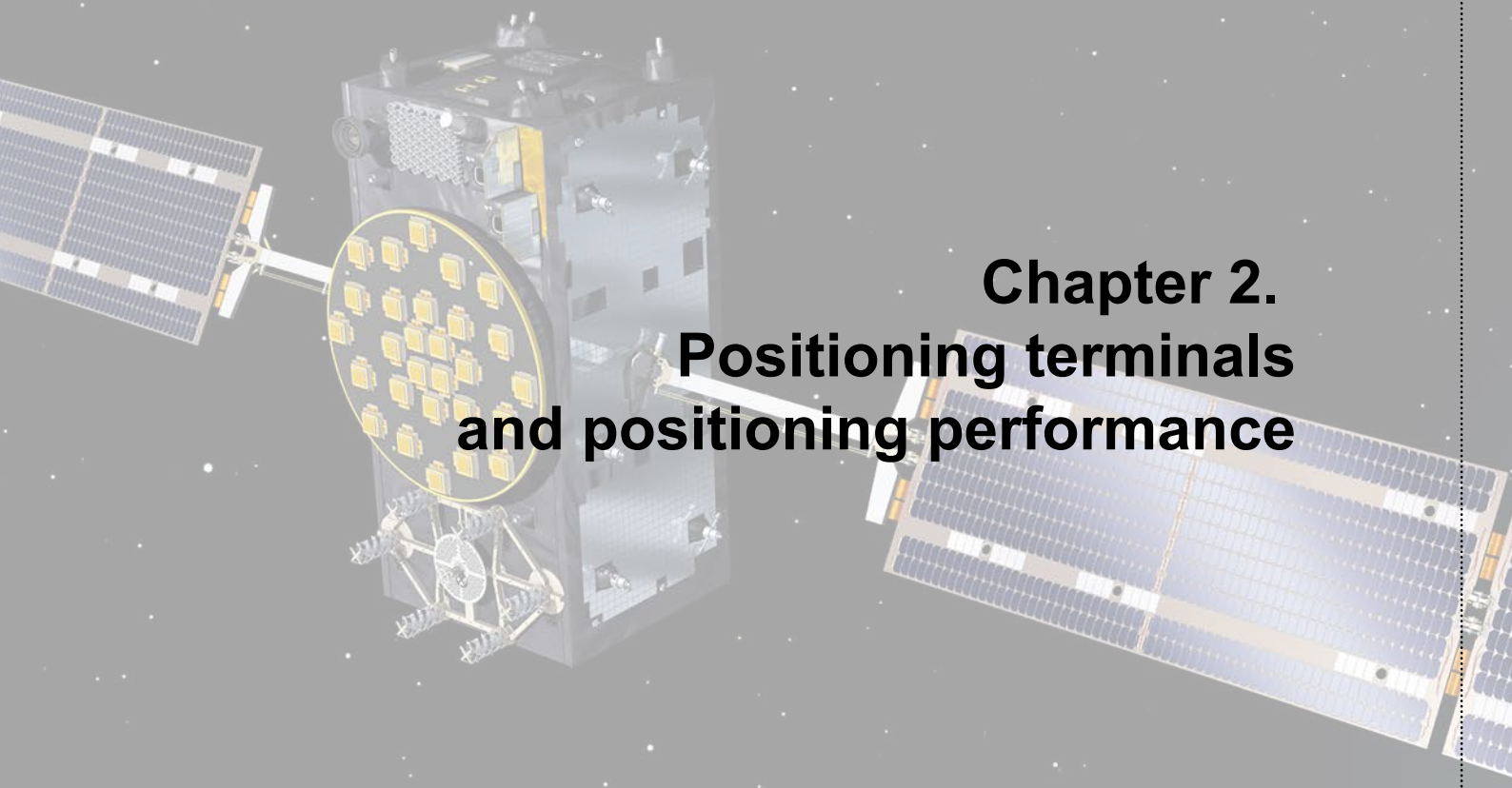
The presentation of four different types of road ITS services demonstrates that the relationship between the performance of an ITS application and that of the positioning terminal is not always direct, requiring a positioning terminal performance derivation process.

There are several reasons for this:

- The raw PVT information is never used directly as it is, but always processed by an application algorithm which can be quite complex, as in the cases of RUC, ICRW and ISA.
- Usually in land transport systems, only the horizontal PVT component is needed by the application but, in some cases, other derivative data are also required such as heading for eCall, or velocity and trajectory curvature for ICRW.

Two observations stem from this:

- Firstly, the current specifications for required positioning performance are mostly either subjective or driven by the performance of the available technologies, without due consideration of how the positioning information is used by the application.
- Secondly, the ITS and smart mobility community needs a standardized and scientifically rigorous method for deriving the performance of the positioning terminal from the E2E performance of the ITS service. Such a method, referred to in this *Handbook* as *Sensitivity analysis*, is presented in the Chapter 3.



Chapter 2.
Positioning terminals and positioning performance

2.1 GNSS positioning

This *Handbook* addresses important issues related to the use of GNSS-based positioning terminals, which can either be pure GNSS receivers or GNSS receivers that are hybridized with other sensors. In any case, the GNSS part will be the most sensitive to the environmental conditions. This why this section and that which follows deal only with this technology.

GNSS are satellite systems providing autonomous, geo-referenced positioning, velocity and timing information with global coverage using radio-wave signals emitted by a navigation satellite constellation [2]. At present, modern society is highly reliant on GNSS, including the GNSS time service.

Currently, there are four operational GNSSs, namely the US NAVSTAR GPS, the Russian GLONASS, the European GALILEO and the Chinese BeiDou (formerly called COMPASS). GPS, GLONASS. Galileo[1] and BeiDou will be fully operational from 2020. The simultaneous usage of satellites from several GNSSs by a single receiver is called "multi-constellation GNSS" and it will provide enhanced performance based on multi-constellation and multi-frequency techniques.

GNSS satellites are mainly located in Medium Earth Orbits (MEO), approximately 20,000 kilometres above the Earth's surface, and continuously broadcast signals. The GNSS positioning principle resides on the trilateration concept by which an unknown receiver location is estimated using distance measurements observed from the known locations of satellites. The basic observable of the system is the time required for a signal to propagate from the satellite to the receiver multiplied by the speed of light in order to compute a distance. Ranges from a minimum of three satellites are required to estimate the user's 3D position. However, as the clocks of the satellites and the receiver are not synchronized, the distance contains an unknown clock offset and is, therefore, referred to as "pseudo-range". As the highly accurate atomic clocks in the satellites are synchronized between each other (satellite clock corrections to GNSS time are broadcast in the navigation message), this offset can be considered as identical for any satellite. Thus, a fourth satellite is sufficient to solve this time unknown and also, by design, to transform each GNSS receiver into a worldwide synchronized time source. Furthermore, GNSS provides the user's velocity via measurement of the Doppler frequency of the received GNSS signals. This frequency is a result of the relative satellite-receiver motion.

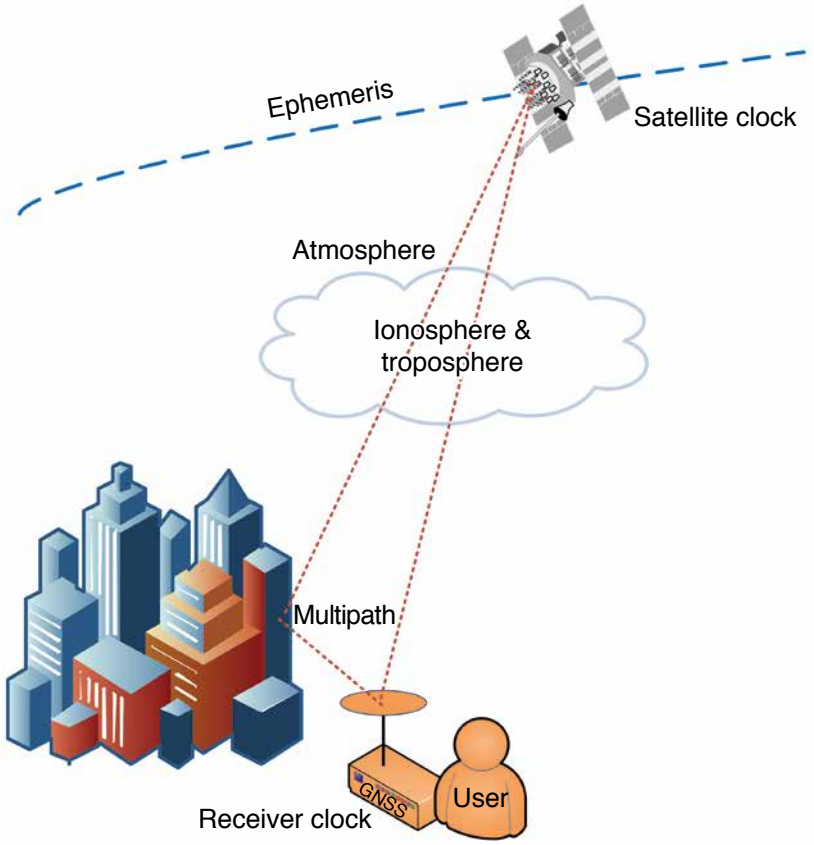
The quality of raw GNSS measurements (also called observables) is affected by several factors originating from the satellites, signal propagation and receiver. Satellite clock offsets and inaccurate orbit information directly bias the pseudo-range measurements. The signal transmitted by a satellite propagates through the atmosphere, where it is subject to (not entirely predictable) delays caused by ionosphere and troposphere media. At ground level, multipath, namely the reception of signals that are reflected from obstacles such as buildings surrounding the receiver, can occur, causing one of the largest errors that is also difficult to model as it is strongly dependent on the receiver environment. The influence of obstacles is the most significant when the GNSS signal from a given satellite arrives at the receiver only indirectly, since it is much more difficult to discard the faulty measurement. This phenomenon is called

1. On December 15, 2016, the European Commission formally announced the start of Galileo Initial Services, the first step towards full operational capability

"Non-Line-Of-Sight" (NLOS). Finally, random errors are encountered at the receiver level due to receiver thermal noise. GNSS errors are discussed in more detail in the following section.

Figure 6 illustrates the main sources of physical errors that degrade the quality of GNSS signals and, therefore, that of the final position output by the receiver.

Figure 6
The physical error sources affecting GNSS signals



The position error resulting from the measurement errors above also depends on the relative geometry between the receiver and the satellites, referred to as "Dilution of Precision" (DOP). Position accuracy is maximized when the directions of the tracked satellites are more uniformly spread around the receiver. The trilateration problem cannot be properly solved if all the satellites employed to calculate the position are aligned (the optimisation problem becomes singular).

The GNSS signal consists of three parts. The ranging code consists of a stream of pseudo-random binary digits which modulates the carrier wave and permits the computation of the signal travel time. The carrier is a sine radio wave carrying the code. Finally, the navigation message contains all the required data for the position computation, i.e. the ephemeris (parameters to calculate the satellite position at a given time), time parameters and corrections for satellite clock corrections, satellite health information, ionospheric model parameters and satellite almanacs necessary for signal acquisition.

2.2 Characterization of the GNSS PVT errors

After introducing general considerations, this section focuses on error models for GNSS positioning, which are particularly necessary in the context of ITS. A methodology for direct identification of a positioning error model is detailed along with an example in Appendix A of the *Handbook*.

2.2.1 Introduction

GNSS PVT solutions result from complex algorithms implemented in the last computing stage of receiver chipsets. The solution is computed from the raw measurements of Doppler frequencies, code and phase pseudo-ranges, and SNRs (Signal-to-Noise Ratios) that have been produced by the previous multi-channel signal tracking stage.

These algorithms, generally described as “navigation” algorithms, encompass ordinary epoch-per-epoch solvers using different optimisers, the common main ones being the least squares method and the Kalman Filter. A wide range of processing techniques exists and can be combined, even in the case of a pure GNSS receiver with no other sensor:

- weighted least-squares, using, for example, SNR or elevation-based weighting
- code only or use of multiple raw measurements (code + phase)
- GPS L1 carrier only or use of multiple carrier waves and multiple GNSS
- single epoch and single solution, or Kalman filtering taking into account a vehicle motion model
- consistency checking of measurements with fault detection and exclusion

To compute the user position, the position, velocity and time offset of the satellites have to be computed in real-time, based on the decoded navigation message carried by the GNSS signals. These satellite PVTs make the determination of the user’s PVT possible.

2.2.2 Error budget

The error budget (cf. figure 6 and Table 5) gathers errors arising from space segment uncertainties (satellite orbits and clocks), ionosphere and troposphere propagation modelling residuals, multipath impact on tracking and tracking noise. The following table indicates the orders of magnitude of the final ranging error resulting from the different error sources [5]

Table 5
Contribution of the error sources to the average range error (standard deviation)

Source	Range error (standard deviation, 1 sigma)
Residual satellite ephemeris and clock errors	0.5 m
Residual ionosphere error (single-frequency)	4.0 m
Residual ionosphere error (dual-frequency)	0.1 m
Residual troposphere error (assuming latitude and season dependent model)	0.2 m
Multipath error (for code measurements, ~100 times less for phase)	⇒ Several tens of m
Tracking noise (for code measurements, ~100 times less for phase)	< 1.0 m

In the navigation algorithm, errors in the range to the satellites in view result in positioning uncertainties. For example, in ordinary least-squares algorithms, the link between ranging errors and positioning errors is directly proportional to the DOP indicator, i.e. the trace of the geometry matrix. The higher the DOP, the greater the impact of ranging uncertainties on positioning uncertainties.

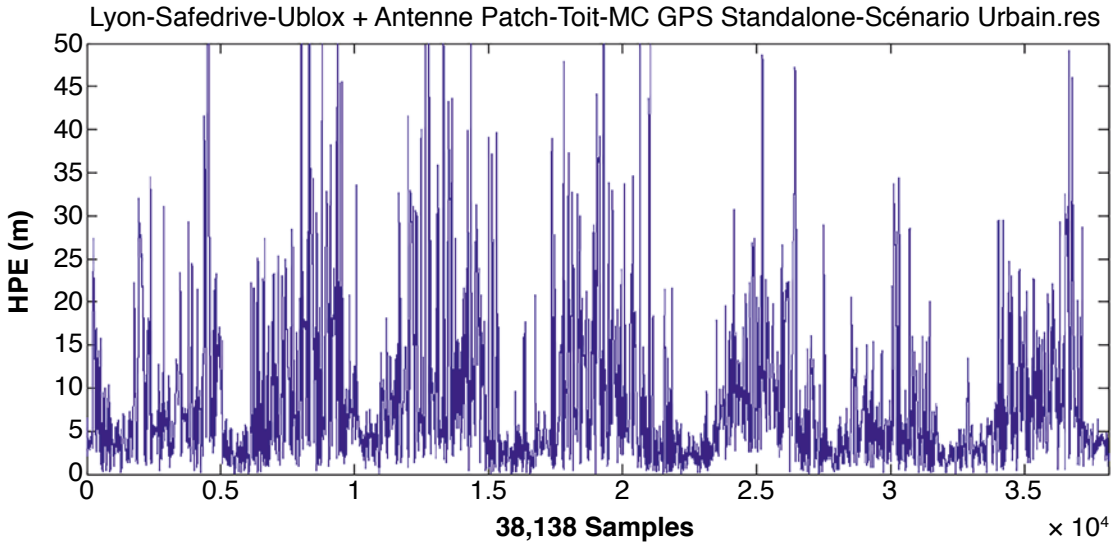
Classically, GNSS position errors are considered as white noise (e.g. in the framework of data fusion processes which perform loose coupling with odometry or inertial measurement unit integration).

However, this assumption is only valid when it is assumed that the pseudo-range errors are white noise and when the position is computed by the least-squares method because of the linearity between position and pseudo-ranges in the least-squares solution. When a Kalman filter is used, the filter algorithm produces positioning errors that are mechanically correlated. Moreover, the white noise ranging error assumption is far from being valid when an NLOS signal is tracked, which is applies particularly in urban environments due to signal blockage by buildings.

For a vehicle driving in an urban environment, when we look at the Horizontal Position Errors (HPE) for instance, the time signature or series appears to show a mix of randomly distributed peaks of errors rather than uncorrelated (white noise), and deterministic shapes correlated with the environment, representing propagation errors — mainly NLOS errors. These autocorrelated errors are either due to autocorrelation in the pseudo-range errors themselves (coming from the environment), and/or from the computation itself, generally Kalman filter based.

This is clearly illustrated by figure 7, showing the HPE signature of a GNSS receiver during a 2.5 hour test in an urban area. The test consisted of seven identical loops. The similarities between the signatures of the different loops are representative of the deterministic (and thus correlated) component of the error.

Figure 7
Example of time-dependent errors of a GNSS receiver in an urban environment



Source: M3 Systems, EGNOS on the road, 2009

2.2.3 PVT error modelling

PVT (particularly position) error modelling is mandatory for a sensitivity analysis to check the compatibility of a given positioning terminal with a given application algorithm in a given environment. Error models obtained from real experiments allow the generation of a high number of simulated but representative trajectories which are necessary for the analysis.

In view of what we have seen in the previous section, it is challenging to model the errors at the level of raw measurements (or observables) and to propagate this model through the navigation algorithm, which is most of the time unknown and always non-linear. The most efficient way is to identify directly a model that matches the real errors observed at the PVT level as well as possible.

As an example, Appendix A presents an HPE model produced by SaPPART for a given receiver model and a given experimental environment.

For other environments and an operational scenario for a given application, other models should be designed or different settings of the parameters of the same model should be found.

The proposed methodology applies to various environments, leading to various models. The methodology can also be extended to the dimensions of velocity and time.

2.3 Positioning metrics

This section provides a summary of the metrics for the performance characterization of the *Positioning terminal* proposed by the CEN-CENELEC standardization organisation [4]. Each performance feature (i.e. accuracy) is quantified by a corresponding metric. Accuracy, availability and integrity features can relate to any output of the GBPT (i.e. horizontal / vertical position, horizontal / vertical velocity, etc.). The examples of metrics given below apply to horizontal position.

2.3.1 Accuracy

Accuracy refers to a statistical characterization of the error in position, velocity or speed with respect to the ground truth. Usually, accuracy relates to the mean and standard deviation of the error distribution it refers to. However, unless the error distribution is known to belong to a given, well characterized, family of statistical distributions (such as the Gaussian family of distributions), these two parameters may not actually capture the error characteristics of interest. In order to provide a statistical characterization of the error, the proposed accuracy metrics are based upon the 50^{th}, 75^{th} and 95^{th} percentiles of the error CDF. The accuracy metric can be broken down into a family of metrics, each describing the metric of error of a particular component (3D vector, horizontal, vertical, cross track, along track, etc.). In this document only one example metric is presented.

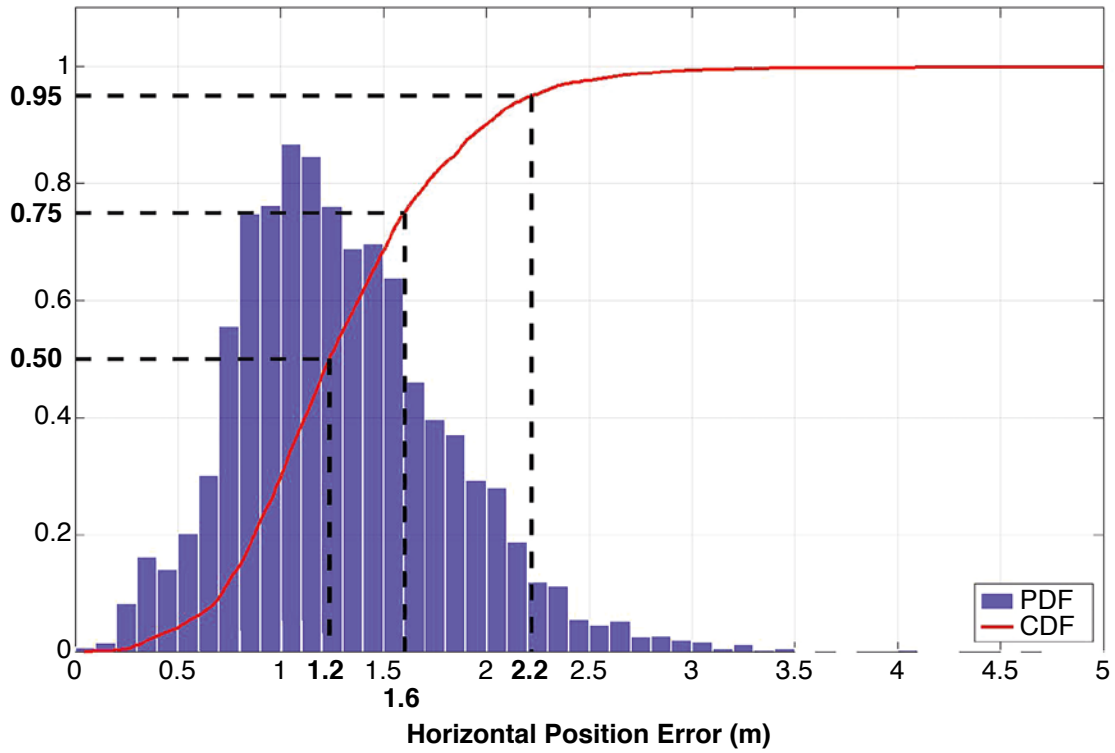
Horizontal Position Accuracy (sample definition)

Horizontal Position Accuracy is defined as the set of three statistical values given by the 50^{th}, 75^{th} and 95^{th} percentiles of the CDF of horizontal position errors.

Figure 8 shows the histogram estimating the Probability Density Function (PDF) and its integral estimating the CDF of the horizontal position error resulting from a real dataset. This dataset was output by a M8N u-blox (multi-constellation EGNOS-capable) receiver operating on the Nantes ring-road, which is representative of a relatively open peri-urban environment.

The 50th percentile, here equal to 1.2 m, represents the median error. It is often used as a metric in the literature and is known as the Circular Error Probable (CEP) at 50%.

Figure 8
Probability density and cumulative distribution function of a real dataset



2.3.2 Availability

Simple availability performance can be defined in terms of the relative amount of time during which the output of interest (whether related to position, velocity or speed) is provided by the *Positioning terminal*. A possible simple availability metric for a particular parameter of interest (in this case the position) could be defined as follows:

Position Availability is the percentage of operating time during which the Positioning terminal provides a valid position output.

A "valid" output is an output delivered by the *Positioning terminal* and flagged "to use", or not flagged "not to use" by the terminal.

However, it may be interesting to know not only the global percentage of time in which a valid position output is available or not, but also how the epochs of availability / unavailability are

Position Availability (sample definition)

Position Availability is the percentage of time intervals of length T during which the Positioning terminal provides at least one valid position output.

distributed over time. In order to account for the distribution of system outages over time, the above definition is generalized to the following that proposes a metric depending on the parameter T:

Typically, depending on the application, the T parameter can be 1 s, 1 mn or 1 h.

Note: it might be interesting, as has been done by the civil aviation community to specify the positioning performance in the different phases of a flight, to choose another definition of availability that considers the epochs when the terminal provides a position output that reaches a certain level of performance in terms of accuracy and integrity. Since this definition depends closely on the application itself and the number of applications is virtually infinite in the ITS sector, this application-based definition of availability has not been proposed in this document.

2.3.3 Integrity

Integrity is a general performance feature which is closely linked to the reliability of the system. It refers to the trust a user can have in the delivered value of a given position or velocity component. This feature is typically expressed by two quantities: the *Protection Level* (PL) and the *Integrity Risk* (IR) associated with it.

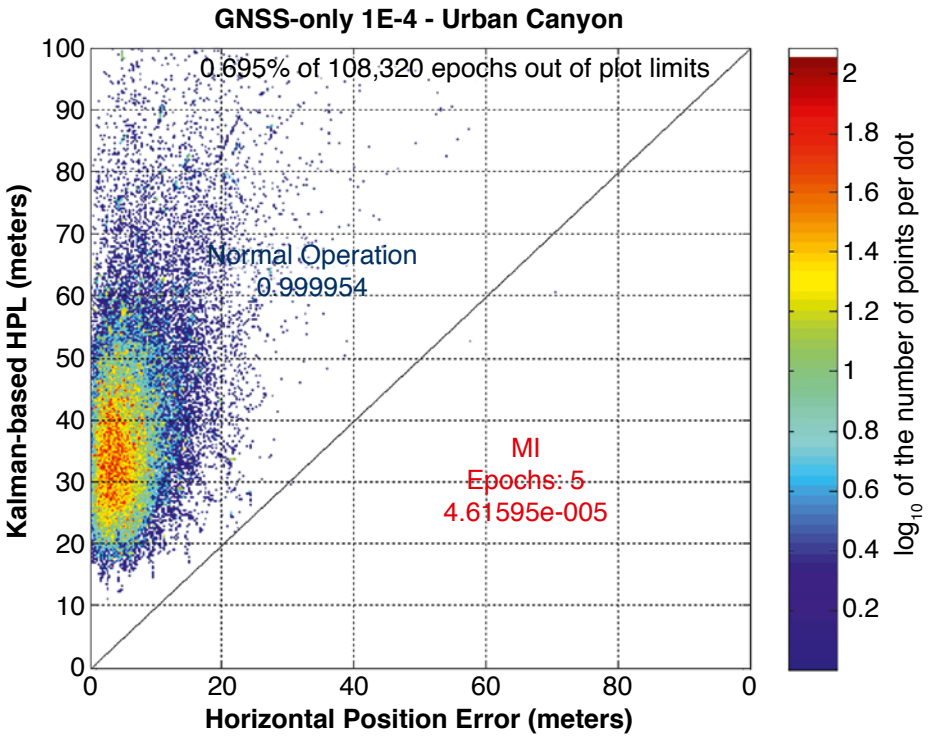
Consequently, integrity metrics make sense only when the output of the positioning terminal includes a *Protection Level* bounding the error in real-time with a given probability.

The metrics adopted by CEN TC5/WG1 and in SaPPART characterize the performance of *Protection Levels* in two different ways. Firstly, they refer to the statistical behaviour of the *Protection Levels* themselves, that is, their size in a statistical sense which is directly linked to their usability for a specific application. Secondly, it describes their metric of reliability as error bounds, referring to the probability (known as the *Integrity Risk*) that a *Protection Level* fails to contain the error.

Naturally, this *Protection level* may not be available at the output of the terminal, for any reason, and these metrics need to be supplemented with a *Protection level* availability metric, similar to the position availability metric.

Figure 9 illustrates the main integrity concepts on a so-called “Stanford diagram”. At each epoch of measurement, a point whose abscissa is the actual position error and whose ordinate is the associated *Protection Level* is plotted. Any point above the first bisector is representative of normal operation (the PL effectively “protects” the user), while a point below is called *Misleading Information* (MI). The ratio of the number of MI epochs divided by the total number of epochs is an estimator of the *Integrity Risk*. For instance, for the test corresponding to the figure, the assessed *Integrity Risk* was ~ $4.6\ 10^{-5}$.

Figure 9
Stanford diagram representing protection level versus position error



Integrity Risk metric

The *Integrity Risk* of a positioning terminal output refers to the probability that the error of an individual metric component (e.g. horizontal or vertical error) exceeds its associated *Protection Level*. In this case, the metric and the definition are merged.

Horizontal Position Integrity Risk (sample definition)

The Horizontal Position Integrity Risk is the probability that the horizontal position error exceeds the horizontal position Protection Level.

Protection Level Performance metric

The statistical characterization of the *Protection Level* operates in a very similar way to the accuracy metrics. The protection level is a positive scalar value that statistically bounds the errors on the quantity estimated by the positioning terminal with respect to a given *Target Integrity Risk* (TIR). The *Protection Level* distribution is usually described using the 50th, 75th and 95th percentiles of its CDF. Importantly, by definition the *Protection Level* percentiles belong to a single-tailed distribution, as *Protection Levels* are always positive real numbers.

Horizontal Position Protection Level Performance (sample definition)

Horizontal position Protection Level performance is defined as the set of three statistical values given by the 50th, 75th and 95th percentiles of the cumulative distribution of horizontal position. Protection levels computed for a certain Target integrity risk (e.g. equal to 10$^{-6)}$.

2.3.4 Timing performance metrics

Timing features are those related to the timing performance of the *Positioning terminal*. For the end user, the key feature is the *Time To First Fix* (TTFF), but also *Timestamp*[2] *resolution*, *Output latency stability* and *Output rate stability* also need to be considered.

Time to first fix is the time the user has to wait for a first valid output after switching on the terminal. For terminals using a GNSS receiver, this time is different depending on the type of start:

- *Cold start* occurs when a receiver is switched on and contains no available information. A full search of the sky for visible satellites must therefore be performed.
- *Warm start* occurs when the receiver is switched on and has a valid almanac (stored or obtained via other means, such as *Assisted GNSS*), and a rough position with approximate information on frequency offset.
- *Hot start* occurs when the receiver is switched on and has both accurate ephemeris and information on frequency offset as well as an accurate initial solution.

Warm TTFF metric (sample definition)

Warm TTFF is defined as the set of three statistical values given by the 50th, 75th and 95th percentiles of the cumulative distribution of the elapsed time from Positioning terminal switch-on in warm conditions until a valid position solution is generated.

Usually, the timestamp accuracy of the *Positioning terminal* meets the requirements of any *Road ITS application*, unless some other system malfunction occurs. However, the following features should be considered by the engineers designing any system integrating a GNSS-based positioning system:

- *Timestamp resolution* is the smallest time lapse which would result in different consecutive timestamps.
- *Output latency* is the time elapsed between the time to which the PVT corresponds and the time at which the same PVT is made available to the *Road ITS application*.
- *Output rate* is the inverse of the time elapsed between consecutive PVT outputs from the *Positioning terminal* (measured in Hz).

While *Timestamp resolution* is itself proposed as a metric, the other two features provide the basis for the definition of two additional features that are called *Output latency stability* and *Output rate stability* respectively and two additional metrics based on the errors as defined below.

Output latency error is the difference between the true output latency and the nominal output latency stated in the *Positioning terminal* specification. The true output latency is computed as the difference

Output Latency Stability metric

Output latency stability is defined as the set of three statistical values given by the 50th, 75th and 95th percentiles of the cumulative distribution of output latency errors.

2. The timestamp is the time that is associated with the PVT output

between the time at which the *Positioning terminal* provides its output message (time of output) and the time to which the PVT information contained in the said output message refers (given by the message timestamp).

Output rate error is the difference between the true output rate and the nominal output rate stated in the *Positioning terminal* specification.

Metric definitions of these two errors are stated for a particular choice of percentiles similarly to the accuracy and *Protection Level* performance metrics.

Output Rate Stability metric

Output rate stability is defined as the set of three statistical values given by the 50th, 75th and 95th percentiles of the cumulative distribution of output rate errors.

2.4 Introduction to performance classes

The metrics defined above form the basis for the establishment of *Positioning terminal* performance requirements. This section introduces a conceptual description of *Performance classes*, with the aim of defining an appropriate positioning-related characterization of the terminal that is both simple and verifiable.

2.4.1 Accuracy classes

The concept is introduced through the example of *Horizontal position accuracy performance classes*. The classification is based on percentile intervals and is best illustrated with CDF plots. It is not intended to identify the actual performance (quantitative figures) of the *Positioning terminal* but only to propose the way the performance requirements should be established or assessed.

For a given scenario, a position terminal will be classified in a certain class according to the results of the measured accuracy metric. In the following example, the two class limits specify that the 50th, 75th and 95th percentiles of the cumulative distribution of horizontal position errors should:

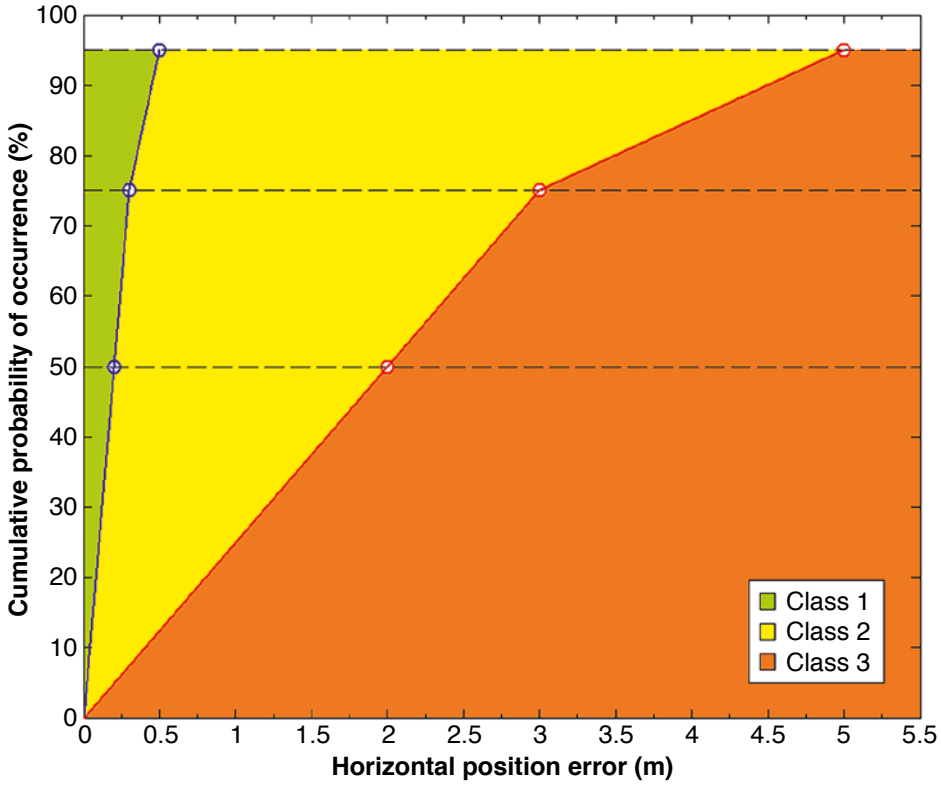
- All be smaller than 0.2 m, 0.3 m and 0.5 m respectively for the position terminal to be categorized as “Class 1 accuracy”.
- All be larger than 2 m, 3 m and 5 m respectively for the position terminal to be categorized as “Class 3 accuracy”.
- All lie between the above limits for the position terminal to be categorized as “Class 2 accuracy”.

This tentative classification is summarized in Table 6 below and illustrated in figure 10.

Table 6
Accuracy performance classification (tentative)

	P = 50th percentile	P = 75th percentile	P = 95th percentile
Class 1	P < 0.2 m	P < 0.3 m	P < 0.5 m
Class 2	0.2 m < P < 2.0 m	0.3 m < P < 3.0 m	0.5 m < P < 5.0 m
Class 3	P > 2.0 m	P > 3.0 m	P > 5.0 m

Figure 10
Accuracy performance classification (tentative)



Source: GMV, GP-START project

Based on the above examples that refer to a classification with respect to the H*orizontal accuracy* feature, the respective classification can also be performed with respect to other performance features as well, as long as the performance metric is expressed in terms of the percentiles of a cumulative distribution. This is the case for all the accuracy metrics, for the *Protection level* performance metric and some timing metrics such as TTFF metrics.

2.4.2 Availability classes

For position availability, a metric which is not expressed in the form of CDF percentiles but only by one indicator (*percentage of time intervals of length T during which the Positioning terminal provides at least one position output*, see 2.3.2), the classes may be described in a simpler way, for instance, for a given T = 10s:

- “Class 1 availability”: percentage > 99%
- “Class 2 availability”: 95% < percentage < 99%
- “Class 3 availability”: percentage < 95%.

Table 7
Availability performance classification (tentative)

Availability Classes	A = Availability (T)
Class 1	A > 99%
Class 2	95% < A < 99%
Class 3	A < 95%

Note: high accuracy and high availability often stand in conflict with one another; a high-grade survey GNSS receiver produces very accurate positions but with low availability in constrained environments while a high-sensitivity receiver outputs fairly low-accuracy position estimates with high availability. Therefore, it is strongly recommended to consider these 2 features together when qualifying or specifying the performance of a GNSS-based positioning terminal.

2.4.3 Integrity classes

Integrity metrics and classes make sense only if the PVT information output by the receiver contains at least a *Protection level* for a given PVT component (e.g. horizontal position), associated with an *Integrity risk*.

In this context, *Protection level* availability classes can be defined in a similar way as for position availability.

Insofar as a *Protection level* and an *Integrity risk* are available for horizontal position, a classification can be proposed for both features as follows:

Table 8
Integrity Risk performance classification (tentative)

IR Classes	Integrity Risk
Class 1	IR < 1E-6
Class 2	1E-6 < IR < 1E-4
Class 3	IR > 1E-4

Table 9
Horizontal Protection Level performance classification (tentative)

	P = 50th percentile	P = 75th percentile	P = 95th percentile
Class 1	P < 1 m	P < 1.5 m	P < 2.5 m
Class 2	1 m < P < 10 m	1.5 m < P < 15 m	2.5 m < P < 25 m
Class 3	P > 10 m	P > 15 m	P > 25 m

2.4.4 Multi-parametric classification

A given GBPT operating in a given environment can be placed in different classes with respect to the different features. We can therefore propose multi-parametric labelling identifying all the relevant performance data at the same time, as in Table 10.

Table 10
Example of multi-parametric labelling of a given GBPT for a particular environment

Horizontal Accuracy	Position Availability	HPL Availability	HPL size	Integrity Risk	…
Class 1	Class 3	Class 2	Class 2	Class 1	…

2.5 Conclusion

A Positioning terminal based upon a GNSS receiver is a complex system whose performance levels are not straightforward to characterize. The factors that degrade the PVT output have many different sources and at the PVT level result in a mix of random errors, which more or less obey standard probability distributions, and deterministic errors that are highly dependent on the operational conditions in which the terminal is used, particularly the environment.

This has two main consequences:

- Firstly, there is no simple model of these errors (which may affect any component of the position, or of the speed, or the timing features) and an appropriate model has to be identified for each operational scenario from real errors obtained during field tests.
- Secondly, credible performance characterization needs to be based upon a detailed assessment or analysis of several performance features, which are sometimes antagonistic, as is the case with accuracy and availability. For the types of performance that are characterized by error distributions, it is recommended to use a metric that represents the entire distribution as well as possible, for example the set of the 50^{th}, 75^{th} and 95^{th} percentiles of the error CDF.

From some specific numerical values of these percentiles, acting as gauges, it is possible to derive performance classes for each performance feature.

Once the terminal is classified with respect to the different metrics, its global performance can be represented by a multi-parametric label that will be representative of its behaviour in a given environment.

Chapter 3. Sensitivity analysis: matching positioning performance to the required E2E performance

3.1 General methodology

As mentioned in the conclusion of Chapter 1, the ability of the Positioning terminal with the application to deliver the required E2E performance to the end user cannot be established easily. For this purpose, a rigorous method should be applied such as the *Sensitivity analysis* described below.

The *Sensitivity analysis* is based upon field tests of the GBPT carried out under real conditions to identify a PVT error model that is effective for the automated generation of a high number of synthetic degraded trajectories to be processed by the application module. These synthetic trajectories are identified as “Degraded PVT” in figure 11 and sometimes called “cloned trajectories”, or, for the sake of simplicity, just “clones”, in the text below.

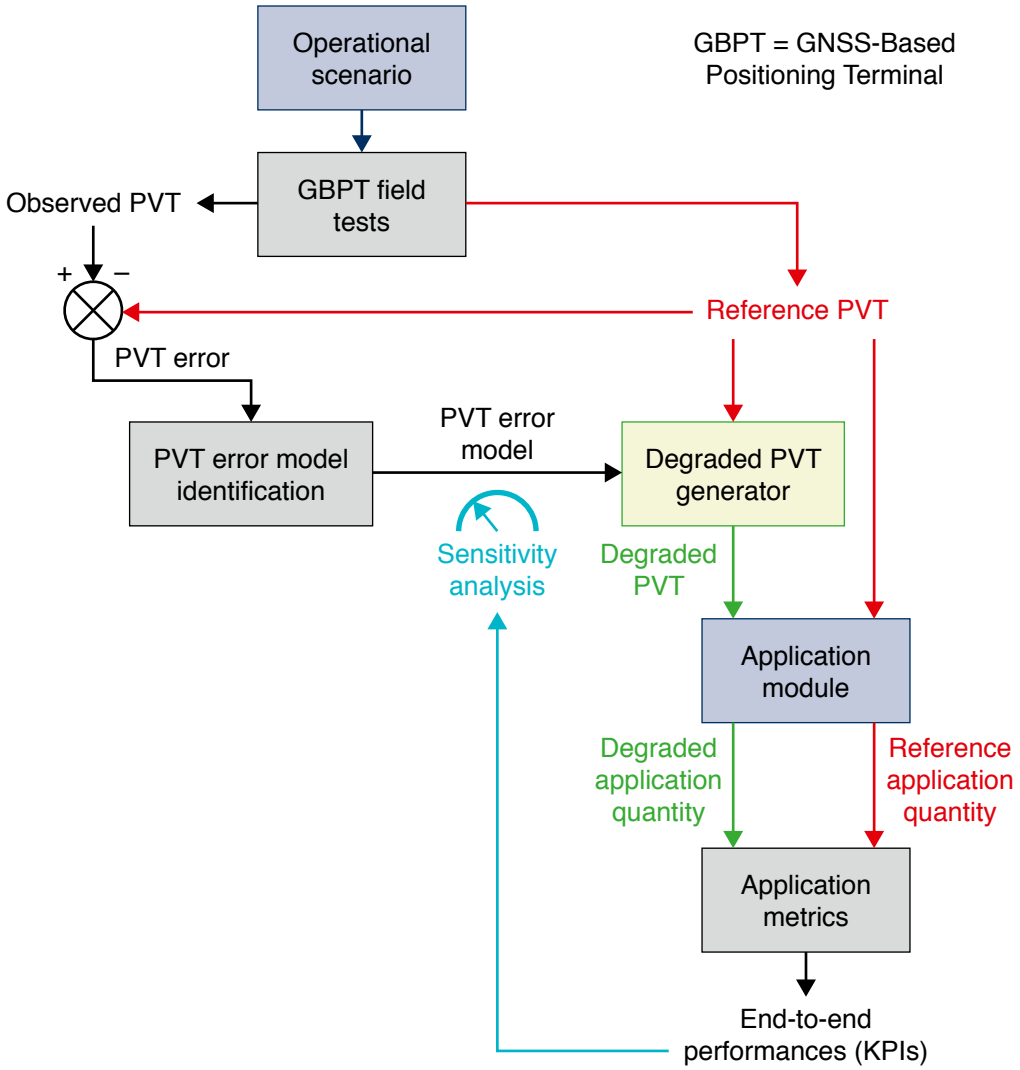
The field tests are executed with a reference trajectory measurement system that is able to deliver a reference PVT that is at least ten times more accurate than the observed PVT from the GBPT under test.

The E2E performance of the system, which depends on the performance of both the GBPT and the application itself, will be assessed from the outputs of the application module using the E2E performance metrics.

This method has the advantage of “multiplying” field tests executed under real operational conditions, which are the only ones capable of capturing the real physical phenomena, in order to run a high number of trials, which is necessary in order to assess performance levels which are generally expressed by low probabilities.

The sensitivity of the system to the performance of the GBPT can be analysed by artificially increasing, step by step, the amplitude of the errors generated by the PVT error model until the point when the target E2E performance is no longer reached. Figure 11 represents the general principle of the Sensitivity analysis.

Figure 11
The Sensitivity analysis method



3.2 Examples of matching in terms of position accuracy

This section presents two different intelligent transport applications, already described in Chapter 1, to which the *Sensitivity analysis* method has been applied in order to analyse the sensitivity of the KPIs to the performance of the positioning terminal.

For each example, the same structure has been applied:

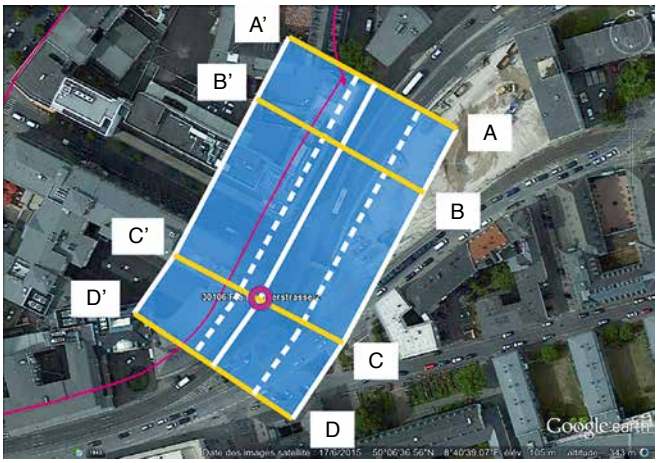
- description of the application and definition of the chosen KPIs
- description of the PVT error model used
- description of the tests performed
- results and conclusion

3.2.1 Road User Charging

Description of the application and its KPIs

The RUC application is based on virtual gantries (VG) defined as geo-objects specified by latitude and longitude. They comprise a tolling point (pink circle on figure 12), a central polyline and crossing segments (AA', BB', CC', DD' in yellow) with left and right extensions of the road (dashed line), and outer tolerances to the left and right (solid line), in metres. The tolerances can be made very large if there are no neighbouring roads. A virtual gantry has a designated driving direction.

Figure 12
Description of a virtual gantry



This virtual gantry mechanism is more complex than the one described in figure 3: when a vehicle passes through the gantry, a score is computed. This score takes account of two parameters. The first is the number of segments crossed (AA', BB', CC', DD'). In figure 12, if the vehicle crosses the 4 segments, this parameter is maximised (i.e. 1). The second parameter is the proximity of the intersection between the trajectory (magenta curve) and the crossing segments: the closer this intersection is to the middle of the crossing segments, the higher the value of the parameter. The final score weights these two parameters in order to give a result of between 0 and 1. A score of 1 means that the virtual gantry has actually been passed. If the score is below a predefined threshold, it is considered that the vehicle has not passed through the virtual gantry.

Data acquired by the SaPPART partner Q-Free in the city of Frankfurt, during the Norwegian national SAVE project, were used for this *Sensitivity analysis*. A first set of virtual gantries was defined manually all along the repeated trajectory in Frankfurt (in green in figure 13). These virtual gantries should be detected, making it possible to determine the number of detections missed by the process. The reference trajectory crosses 20 virtual gantries. Additional virtual gantries (in orange) were added in the vicinity of the first set. The aim of this second set was to create false detection candidates and make it possible to quantify false detections. Some of these additional virtual gantries belonged to other streets, but some were also totally false. They were located in particularly challenging areas in order to trigger false detections. Thus, the false detection rates observed in this example do not in any way reflect those of a real system, and are much higher than what is usually acceptable.

Figure 13
Virtual gantries designed in an experimental area in Frankfurt

Figure 14 and figure 15 present examples of missed detection and false detection respectively

The following colouring scheme has been applied:

- green curve: reference trajectory
- blue curve: real trajectory of the GNSS (u-blox 6T) receiver
- magenta curve: one sample of cloned trajectory
- blue virtual gantry: virtual gantry in the original Q-Free database
- orange virtual gantry: virtual gantry added especially for false detection purposes.

Figure 14
An example of missed detection

Figure 15
An example of false detection

Global KPIs for missed detections and false detections have been chosen with the agreement of Q-Free and according to [6] and the SaPPART *White paper* (Table 3). The KPIs are the following, for the N clones in the u-blox-6T trajectory:

- Correct Charging Rate (CCR):

$$CCR = \frac{TotalNbCorrectDetections}{N \times RefNbCorrectDetection}$$

- Over Charging Rate (OCR):

$$OCR = \frac{TotalNbFalseDetections}{TotalNbChargingEvents}$$

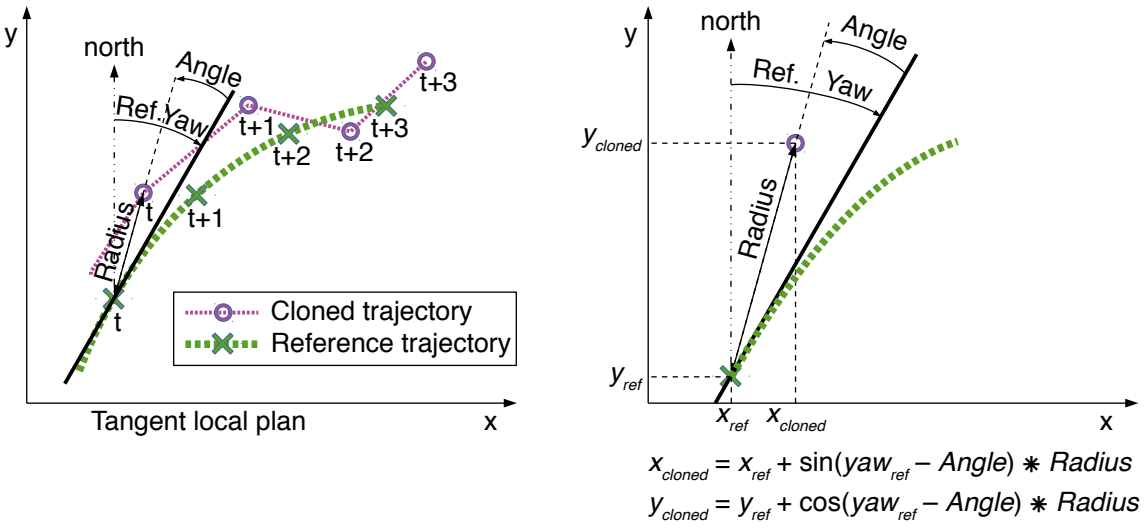
Where N denotes the number of cloned trajectories, *TotalNbCorrectDetections* is the expected number of correctly detected virtual gantries out of all the N cloned trajectories, *RefNbCorrectDetections* is the expected number of correctly detected virtual gantries with the reference trajectory, *TotalNbFalseDectections* is the additional number of orange virtual gantries that were detected when they should not have been, and *TotalNbChargingEvents* is the sum of *TotalNbCorrectDetections* and *TotalNbFalseDectections*.

Other KPIs dedicated to each virtual gantry could be defined. However, only these two global KPIs are considered in this *Handbook*.

Description of the PVT error model used

The PVT error model used is presented in Appendix A. Real 2D position errors and angle errors were “cloned”, and these clones were combined to produce “cloned trajectories” representing probable trajectories that could be produced by the real positioning terminal in operation in the considered environment (in our case Frankfurt city centre):

Figure 16
Reference trajectory and cloned trajectory



$$x_{cloned} = x_{ref} + \sin(yaw_{ref} - Angle) * Radius$$
$$y_{cloned} = y_{ref} + \cos(yaw_{ref} - Angle) * Radius$$

The reference trajectory was used, and cloned points were computed by adding cloned radii and angles onto the reference points (figure 16).

Description of the tests performed

Five different scenarios were identified corresponding to five different degradation levels of the cloned trajectories, namely: 22 m, 44 m, 66 m, 88 m and 176 m.

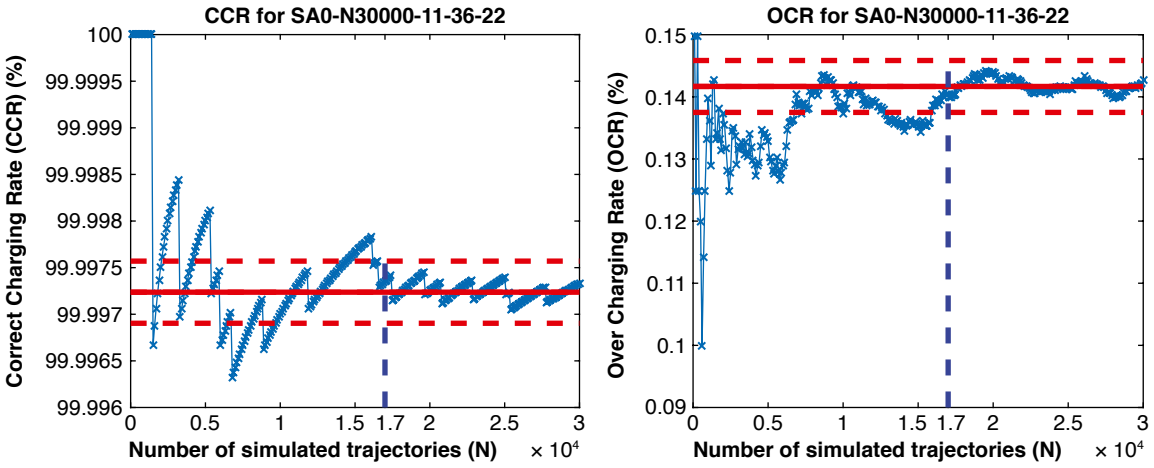
These parameters correspond to maximum value of the truncated Laplace distribution that models the step height probability for the radius component of the model (see Appendix A – Table 12: Simulation parameters). For instance, 44 m means that the stepwise component of the random error on the horizontal position is drawn from a truncated Laplace distribution whose maximum is 44 m. Given the shape of the distribution, such errors will be rare but may be encountered from time to time.

It is important to note here that 22 m is the level that was identified from the experimental data from the u-blox receiver. The other levels were levels that had been intentionally increased for the study and that are representative of less accurate receivers.

a) Fixing the number of clones

The first parameter to set was the number N of simulated trajectories we had to produce in order to be confident that the computed metrics were representative of the *Sensitivity analysis* set. To produce representative metrics, a compromise must be found between the time of computation and the number of simulated trajectories.

Figure 17
Change in CCR (a) and OCR (b) over 30,000 runs



a) b)

Figures 17 represent the respective changes in CCR and OCR during 30,000 simulation runs. CCR and OCR have been quantified in terms of percentages. The full red line corresponds to the mean of the last third of the runs (from 20,000 to 30,000); the dashed red lines correspond to +10% and -10% of the highest oscillation. As we can observe, from N=17,000 the oscillations stay within the 10% bounds. This value of N=17,000 was used for the next phase of RUC assessment for the 5 different scenarios.

b) Injection of the ground truth in the RUC algorithm

In order to compute the expected results, we need to inject the ground truth into the RUC algorithm. The output will be used as the reference for analysing the deviation from the output of the cloned trajectories.

c) Injection of the whole set of cloned trajectories into the RUC algorithm

The same process as in (b) above but with the clones of the original u-blox trajectory.

d) Comparison of each output from the processed clones with the reference outputs

Each output from the RUC algorithm of the cloned trajectory is then compared with the reference output and all the false detections and missed detections are counted. These two metrics are needed to compute the KPI.

e) Computation of the KPIs

The final global KPIs can now be computed for each set of degradations.

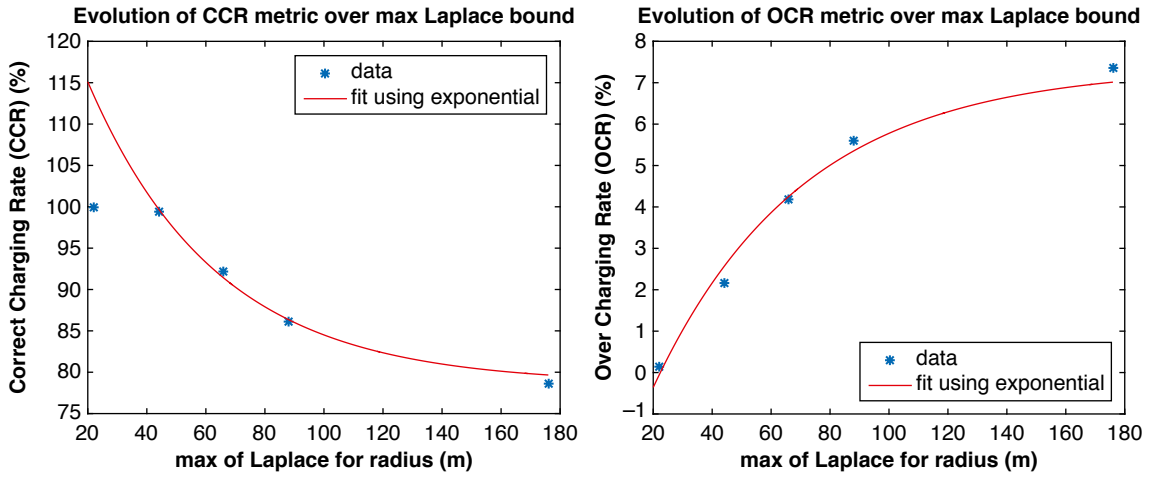
Results

The two KPIs (CCR and OCR) were computed for the 5 different sets of degradation, namely SD-xxm, where xx represents the maximum level of degradation for the clones (also identified as “max of Laplace for radius” on the following figures).

Table 11
RUC Key Performance Indicators for increasing degradation levels

Set of degradation	Total number of missed detections	CCR (%)	Total number of false detections	OCR (%)
SD-22 m	9	99.99	480	0.14
SD-44 m	2062	99.39	7505	2.17
SD-66 m	26631	92.16	13651	4.17
SD-88 m	47105	86.15	17360	5.60
SD-176 m	72672	78.63	21265	7.37

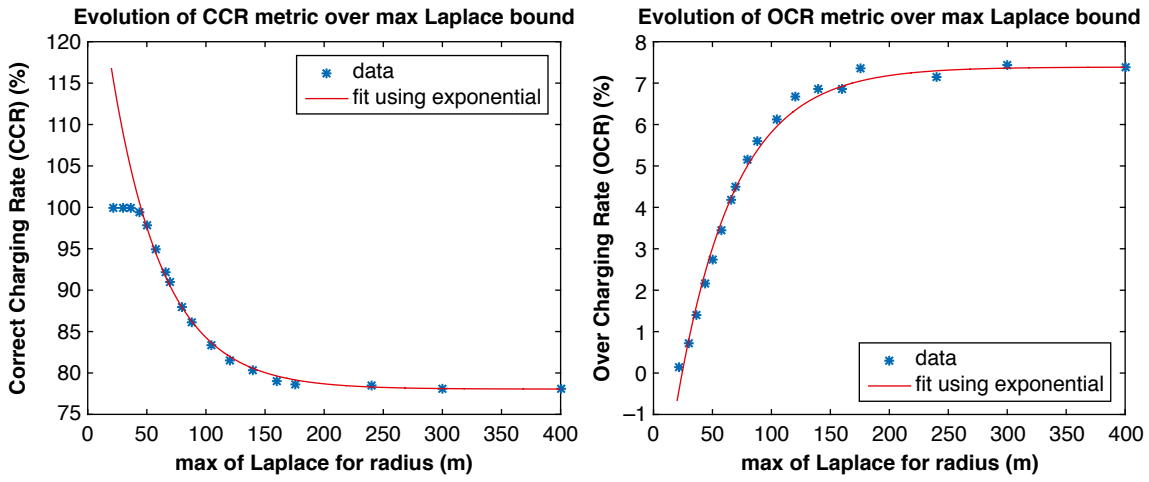
Figure 18
CCR (a) and OCR (b) for SD-22m, SD-44m, SD-66m, SD-88m, SD-176m



a) b)

Figure 18 shows the CCR and OCR computed for 5 degraded trajectories (from SD-22 m to SD-176 m) fas percentages. We can observe an exponential trend of the evolution of CCR and OCR over the Laplace upper bound. In order to confirm this trend, other runs have been realized with higher degradation levels. The exponential trends are confirmed (figure 19).

Figure 19
Asymptotic trend in the RUC metrics for CCR (a) and OCR (b)



a) b)

Conclusions

The *Sensitivity analysis* method applied to RUC provides interesting results, even if the simulation scenario corresponds to a case that will never be encountered in reality.

- For the Correct Charging Rate, the results show that this KPI remains high and stable for the first two levels of degradation but follows a negative exponential curve from the third.
- For the Over Charging Rate, the same negative exponential trend is observed, but it is inverted this time, right from the first levels of degradation.
- In our simulation, the two KPIs converge towards the asymptotic values of 76% and 7.5% respectively when the level of degradation is artificially increased to extreme levels that will never be observed in reality. This tends to prove that the whole system (i.e. the virtual gantries and the RUC algorithm) achieves the minimum performance level, whatever the receiver. This level of performance is actually guaranteed, even if the experienced trajectories are all very scattered.

3.2.2 eCall

Description of the application and its Key Performance Indicators

The objective of this section is to analyse the impact of positioning terminal PVT errors on the performance of the eCall application. As eCall is quite a complex ITS application in terms of the processes and sub-modules involved, and not all the modules/components are influenced by PVT errors, it was decided to simulate only the map-matching that is performed at the PSAP in order to position the vehicle on the road network. Therefore, the simulation of the eCall application was centred on a map-matching process performed in QGIS[3] software, which is an easy-to-use and recognised geographical processing software. In order to determine the sensitivity, KPIs were defined for eCall performance. Map-matching was performed for 5 levels of degradation of the reference PVT and in each case, the KPIs were calculated.

3. http://www.qgis.org/en/site/; a free and open source geographic information system

Based on the availability of measured GNSS positions and the PVT error model developed in SaPPART, the geographical area selected for the simulation was a certain part of the urban road network in Frankfurt, Germany. This in turn determined the decision to select two operating environment scenarios ─ "junctions" and "close parallel streets" ─ in order to perform our eCall sensitivity analysis.

QGIS was used to perform map-matching based on nearest neighbour analysis, i.e. for each reference point and each "cloned" point[4] the nearest point on the road network was identified. The results are exported and further processed to calculate the KPIs. This relatively simple map matching algorithm was chosen because it is a standard, well implemented, tool available in QGIS. More advanced algorithms exist, however they are not implemented as standard in QGIS nor do they exist as an easily accessible, non-proprietary, plugin. In addition, the very restricted PVT information included in the eCall message is generally not enough for the requirements of advanced map-matching algorithms. Therefore, if a complex map matching algorithm had been used, we would not have achieved a realistic simulation of the actual behaviour of the eCall application.

The following KPIs have been defined:

a) Percentage of exact positioning (KPI1)

This KPI measures how many cloned points are map-matched to the same points of the road network as the corresponding reference points. The concept is explained below.

Note: The algorithm used in QGIS works in a slightly different way from the standard map-matching process described in 1.2. Firstly, we created points on the road segment that are evenly spaced about 1.1 m apart (this value was selected after several trials). Then, map-matching is performed between the reference/cloned points and the points create on the road network. So the QGIS algorithm does not project a point, instead, for each reference/cloned point, it actually finds the closest point (in distance terms) out of those created on the road network. In this way, a cloned point that is spatially close to the reference point is matched to the same point as the reference point. This method is equivalent to using the standard map-matching process to find the cloned points that are matched on the same road segment and within a certain radius from the matched position of the reference point.

KPI1 is calculated individually for each reference point as follows:

$$KPI1 = \frac{\text{Number of clones matched to the same road point}}{\text{Total number of clones for the reference point}} \times 100$$

b) Percentage of correct road positioning (KPI2)

This KPI measures whether or not the position of the cloned point is matched tithe correct road/street, regardless of the longitudinal distance (along the road) from the matched position of the reference point on that road. Points positioned on the same road include those that are matched to the same position as the corresponding reference point. The correct road is defined as the street with the same name as the street where the reference is matched. To ensure that this rule is followed, all available road network segments are checked and if needed, further processing was done. For example, if a straight road segment crosses a junction and continues afterwards with a different street

4. A "cloned" point is a point on a "cloned" trajectory that corresponds to the point on the reference trajectory that is considered.

name, then that segment is split into two segments. Similarly, if several separate straight segments have the same street name they are joined. Care is taken to keep streets with the same names but with one-way traffic operating in different directions as separate segments. The reasoning behind this methodology is that, in an urban environment, it can be considered acceptable for a PVT error to result in matching at a certain distance on the same street, but it would not be acceptable to match to a street with a different name because this would confuse the rescue teams.

KPI2 is calculated individually for each reference point as follows:

$$\text{KPI2} = \frac{\text{Number of correct road positions}}{\text{Total number of clones for the reference point}} \times 100$$

c) Average absolute error of the longitudinal positioning along the road (KPI3)

For the cases with correct road positioning, the longitudinal error (LE) in metres with respect to the matched position of the reference point is calculated. The arithmetic mean of the values is then calculated. Points that are matched to the same position as the corresponding reference point are not considered. The processing software outputs both the LE for each cloned point as well as KPI3.

KPI3 is calculated individually for each reference point as follows:

$$\text{KPI3} = \frac{\sum_{i=1}^{N} |\text{LE}_i|}{N}$$

where *N* is the number of cloned points with correct road positioning and LE≠0.

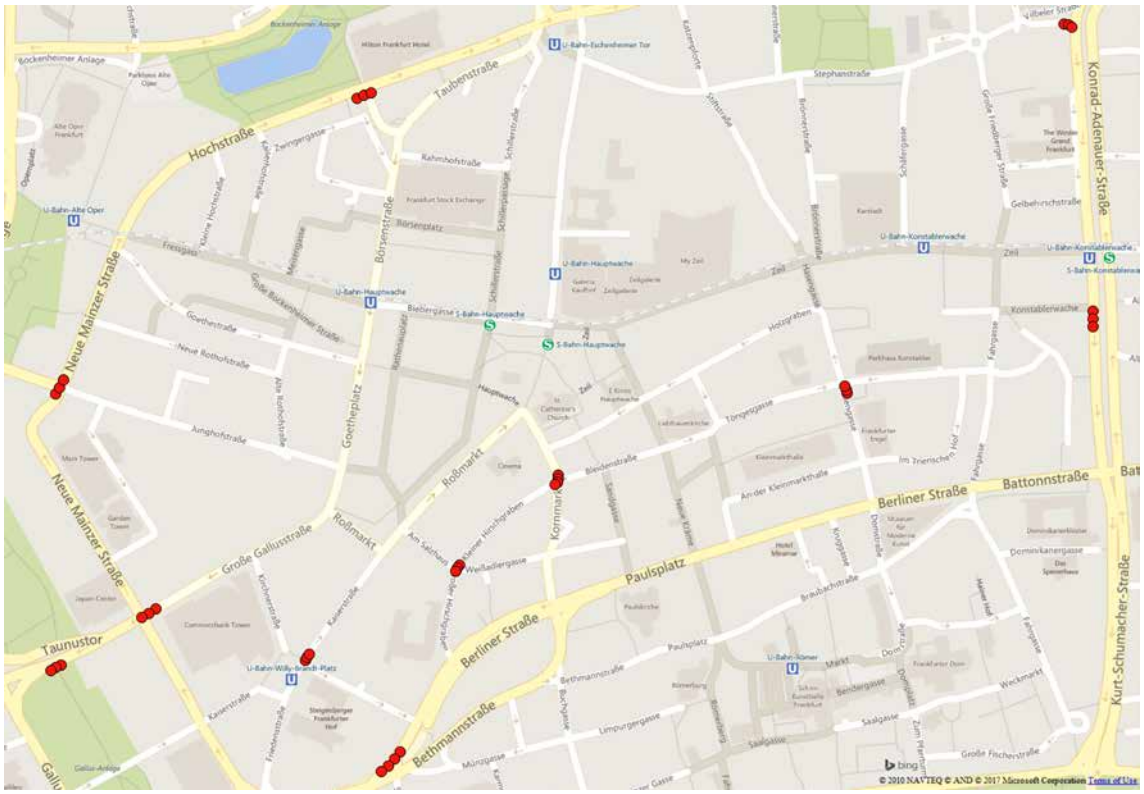
Description of the PVT error model used

The methodology used in the analysis is to simulate the eCall application and use a PVT error model in order to generate positioning errors to be fed into the simulation. The PVT model used here was the same as that used for RUC, i.e. the model developed by SaPPART and presented in Appendix A.

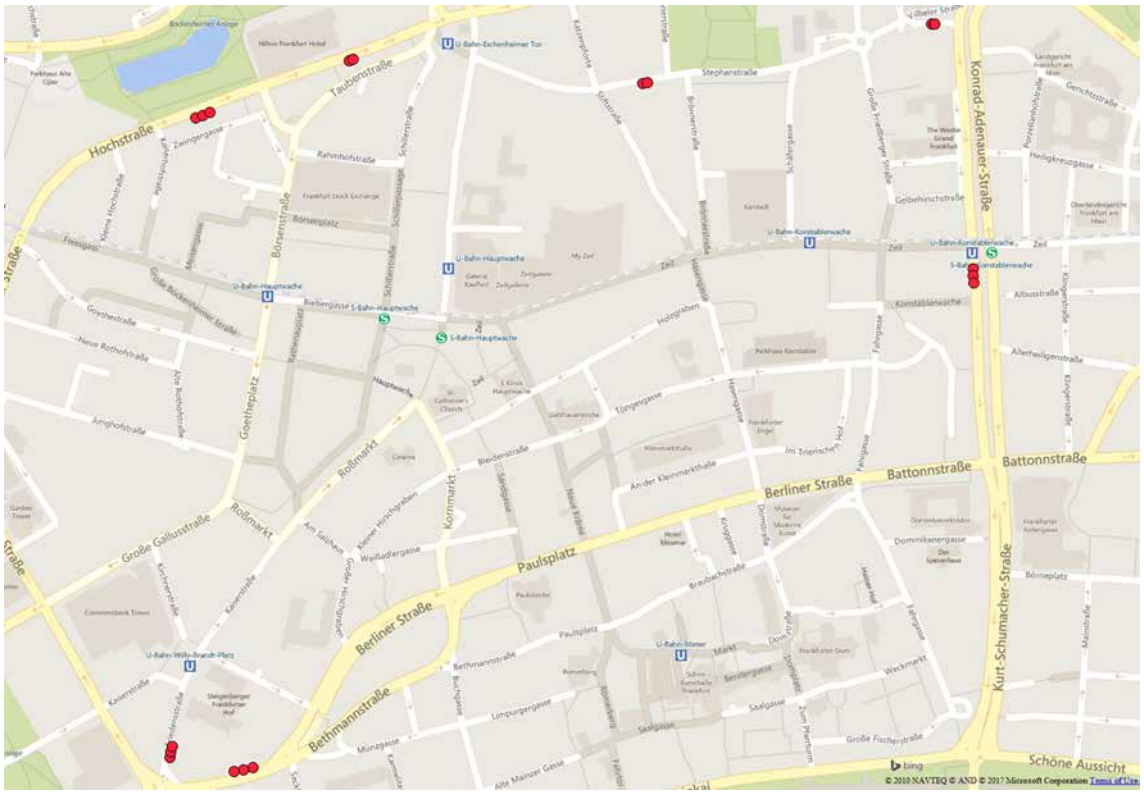
The following main data are required for the study:

- A set of reference points. These are selected from the reference tracks of measurements acquired in Frankfurt and made available by Q-free for the development of the SaPPART PVT error model. Between two and four reference points for each “junctions” and “parallel roads” scenario were chosen.
- Figure 20 shows in red the reference points chosen for the “junctions” and for the “parallel roads” scenarios. There are 11 junctions each with 3 – 4 closely positioned points. Similarly, there are 7 parallel roads locations each with 2 – 4 closely positioned points.

Figure 20
Reference points for the “junctions” (a) and the “parallel roads” (b) scenarios



(a) “junctions” scenario



(b) « parallel roads » scenario

- Cloned points for each selected reference point, representing different levels of degradation of the original reference position. A software routine written in LabView was used to extract from the files that record the cloned trajectories the points matching the timestamp of each selected reference point. 17,000 cloned points were used for each reference point.
- The road network of the Frankfurt test area. Relevant lines in this road network around the selected reference and cloned points were processed in order to create successive chains of points along them that were evenly spaced approximately 1.1 metres apart. These points are needed for the map-matching algorithm.

Description of the tests performed

As mentioned above, two operational scenarios were used: “road junctions” and “parallel roads”. A total of 35 reference points were selected for the former and 18 for the latter. For each reference point, 17,000 cloned points were used.

The KPIs were calculated for each scenario and for each 5 degradation levels of the reference PVT: 22 m, 44 m, 66 m, 88 m and 176 m (see 3.2.1.3 and Appendix A for the meaning of the degradation levels).

The methodology consisted of the following major processing steps:

a) Extraction of text files of cloned points for each selected reference point. Individual files were created for each scenario and each degradation level, resulting in 5 files per scenario. Every file contained 595,000 points (GNSS coordinates) for the “junctions” scenario and 306,000 points for the “parallel roads” scenario

b) Importation in QGIS of the files of the cloned points, the reference points and the points on the road network. Map-matching was performed both for the reference points and the cloned points. In the case of reference points, map-matching was performed only once per scenario and then used for each degradation step

c) Exportation of the results of map-matching from QGIS in three files:

- a list of cloned points with the ID of the matched road point
- a list of reference points with the ID of the matched road point
- a list of road points including their GNSS coordinates and individual, unique, IDs

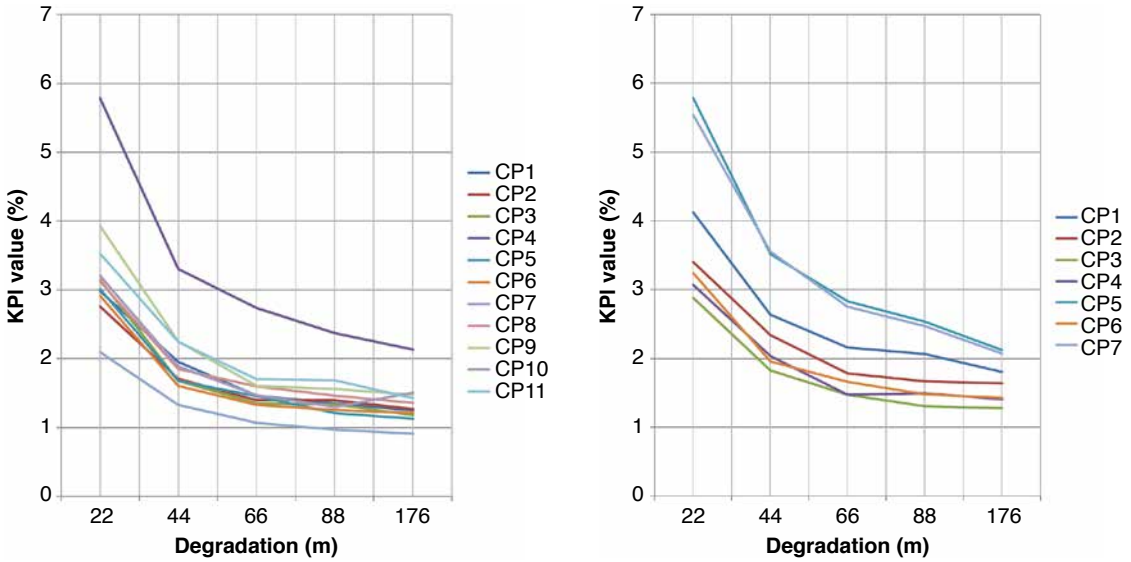
d) Calculation of the KPIs with a specific Excel piece of software.

Results

Percentage of exact positioning (KPI1)

For each set of closely positioned reference points which share the same geographical location, an average KPI1 value is presented in the figure 21 below for both scenarios. Each set is labelled CP1 to CP11 for “junctions” and CP1 to CP7 for “parallel roads”.

Figure 21
Results for percentage of exact positioning for the “junctions” (a) and the “parallel roads” (b) scenarios



a) “Junctions” scenario

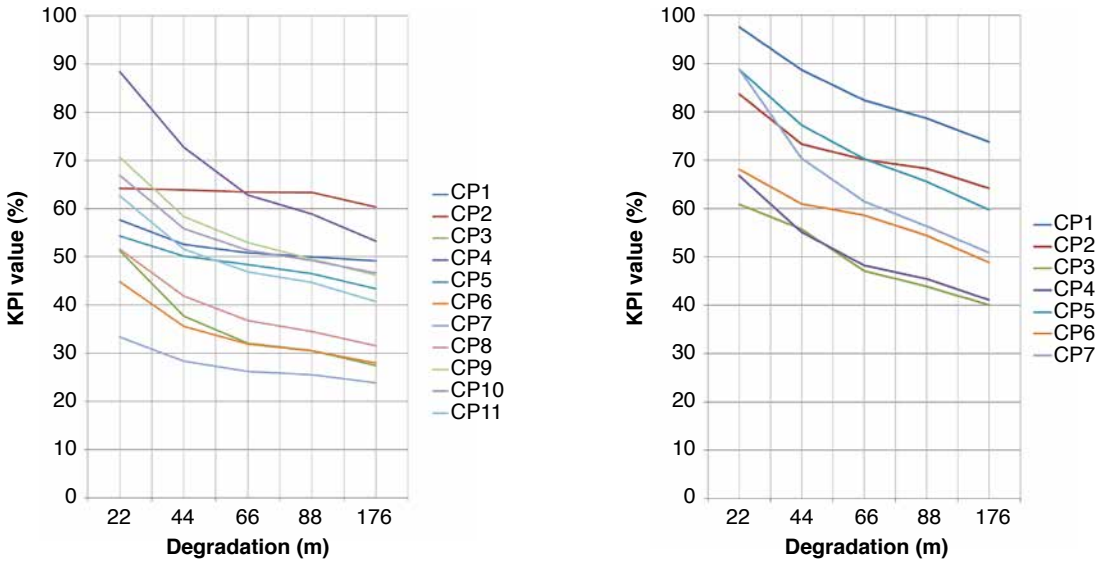
b) “Parallel roads” scenario

It can be seen that the values of this KPI are very low, most likely as a result of the simple, distance-based, map-matching algorithm. Another somewhat expected result is that KPI1 is highly dependent on the geographical layout of the position as there are points with values around 2% and others with values close to 6% for the same degradation. A third conclusion suggested by the analysis is that KPI1 degrades most between 22 m and 44 m, with subsequent values remaining almost stable. The same conclusions as for the “junctions” scenario seem to apply to the “parallel roads” situation.

Percentage of correct road positioning (KPI2)

For each set of closely positioned reference points which share the same geographical location, an average KPI2 value is presented in figure 22 below, for both scenarios. Each set is labelled CP1 to CP11 for junctions and CP1 to CP7 for parallel roads.

Figure 22
Results for percentage of correct road positioning for the “junctions” (a) and the “parallel roads” (b) scenarios



a) “Junctions” scenario

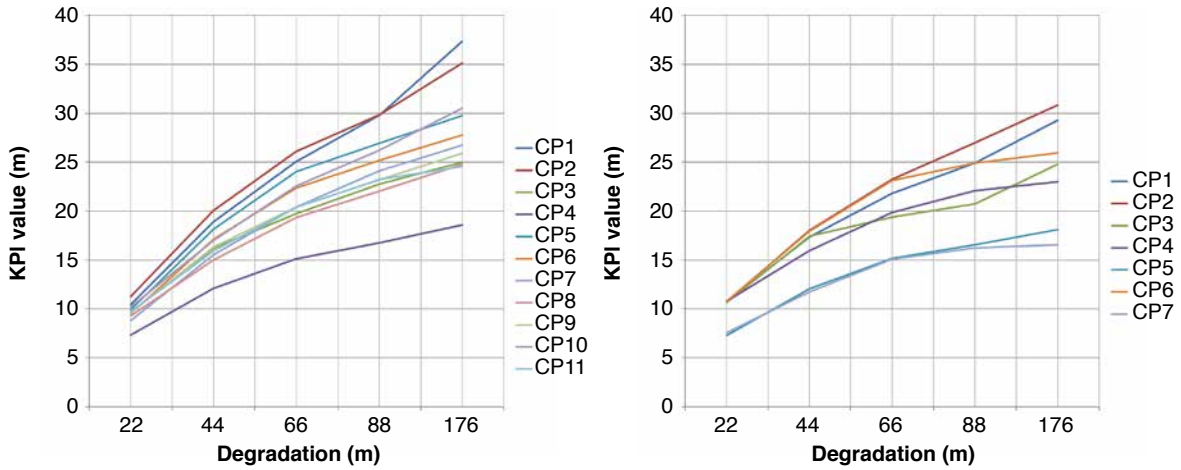
b) “Parallel roads” scenario

The results of the sensitivity analysis show that KPI2 is also dependent on the configuration of the road network where the point is located. Also, the major changes are again between a degradation of between 22 m and 44 m but not so pronounced as for KPI1. In the “parallel roads” scenario, the values exhibit a much clearer descending trend with increasing PVT degradation. Also all values are much higher when compared to those for the “junctions” scenario. This is probably because points have been deliberately chosen far from road junctions and it seems that the PVT errors produce more points that are matched to the wrong road around a junction than elsewhere.

Average of longitudinal positioning error along the road (KPI3)

For each set of closely positioned reference points which share the same geographical location, for both scenarios, an average of their KPI3 values is presented in figure 23. Each set is labelled CP1 to CP11 for junctions and CP1 to CP7 for parallel roads.

Figure 23
Results for average longitudinal positioning error fo for the “junctions” (a) and the “parallel roads” (b) scenarios



a) “Junctions” scenario

b) “Parallel roads” scenario

The results of the analysis show that the KPI3 values are less dependent on the reference point than those of the other KPIs, especially at low levels of degradation. All of them exhibit, as expected, a steady increase as the degradation level rises. The values can be considered acceptable for real life situations in an urban environment since none of them is higher than 37 m. At this distance, considering how correct roads were defined, it can be argued that it would be easy for the rescue teams to identify the real location of an accident. Once again, the results for the “parallel roads” scenario are better than those for the “junctions” scenario.

Conclusions

This sensitivity analysis performed on the eCall application, in two scenarios that are particularly relevant for the application and fairly representative of reality, provides a number of interesting results.

- In terms of correct street detection, measured by the KPI2 indicator, the performance depends significantly on the configuration of the site, especially for the “junctions” scenario, with the ratio between the worst and best scenario varying between a factor of 2 to 3. KPI2 also indicates that this performance is globally quite poor, especially for the “junctions” scenario again, the results being as bad as 32% for the worst “junction” case and 60% for the worst “parallel road” case, at the 1st level of degradation. This performance deteriorates quite rapidly with the degradation level. This poor performance, observed here in an urban environment, can raise important issues for this safety-related application in which the rapidity with which the rescue team reaches the location of the accident is highly critical.
- In terms of longitudinal error, when the street is correctly matched, the values of KPI3 are far more acceptable, varying between 7 m and 12 m for the lowest level of degradation and between 18 m and 35 m for the worst. These numbers are totally acceptable for a rescue team arriving in the correct street.
- The very poor results achieved by KPI1 should not be taken into account since the exact matching rate is not a realistic indicator and was only considered in our study in order to provide a better understanding of the map-matching process.

Overall conclusion

The first deliverable of SaPPART, the *White paper*, explained the role of positioning systems in transportation and the need to correctly assess their performance. It introduced the fundamentals of positioning systems with a particular focus on GNSS. It described the positioning terminal architecture and the parameters used for the characterization of performance.

This *Handbook* goes further by explaining the issue of matching the performance of the positioning terminal to the E2E performance of the road transport system as a whole. After having analysed four different types of road ITS applications with regard to positioning information, the document highlights the main factors affecting the quality of GNSS-based positioning, proposes a comprehensive methodology for modelling the errors in the outputs of the positioning terminal and introduces some basic definitions of performance metrics and performance classes for the positioning terminal. The final and central part of the *Handbook* demonstrates the *Sensitivity analysis* method using two examples of road applications, illustrating the sensitivity of the application KPIs to the quality of the positioning information (in this case, the accuracy).

An obvious conclusion of this document is the absolute necessity of being able to characterize rigorously the performances of the positioning terminals that are on the market and that ITS engineers are thinking of using for developing their systems. Rigorous characterization not only needs clear and applicable metrics, which are currently proposed by standardization bodies, but also the definition of appropriate performance classes and test procedures, which must be done prior to the setting up of any pan-European certification framework. These topics will be addressed in the final deliverable of the SaPPART COST Action, called *Guidelines*, which will be produced at the end of the Action.

References

[1] F. Peyret, P.-Y. Gilliéron, L. Ruotsalainen and J. Engdahl. COST TU1302- SaPPART White Paper - Better use of Global Navigation Satellite Systems for safer and greener transport, 1: TMI. IFSTTAR, Lyon, France, 978-2-85782-706-1, Technique et Méthodes 1, 2015.

[2] E. Kaplan and D. Hegarty, Eds., Understanding GPS Principles and Applications. Norwood, MA, USA: Artech House, 2006.

[3] W. Lewandowski; J. Azoubib; W.J. Klepczynski, GPS: primary tool for time transfer. Proceedings of the IEEE, Vol. 87, No. 1. 1999.

[4] CEN-CENELEC EN 16803-1: Use of GNSS-based positioning for road Intelligent Transport Systems (ITS). Part 1: Definitions and system engineering procedures for the establishment and assessment of performances, ICS 49.140, October 2016

[5] Groves, P. D. (2013). Principles of GNSS, inertial, and multisensor integrated navigation systems. Artech house.Groove's Book

[6] ISO/TS 17444:2017 "Electronic fee collection - Charging performance

Part 1: Metrics

Part 2: Examination Framework"

[7] M. Ortiz, D. Bétaille, F. Peyret, O.M. Lykkja, S.-P. Oseth, Assessment of end-to-end performances of a GNSS-based road user charging system, ENC-GNSS 2016, Helsinki, June 2016

[8] M. Hutter, "Exact Bayesian regression of piecewise constant functions", Bayesian Analysis, Vol. 2, No. 4, 2007, pp. 635–664

[9] Regulation (EU) 2015/758 of the European Parliament and of the Council of 29 April 2015 concerning type-approval requirements for the deployment of the eCall in-vehicle system based on the 112 service and amending Directive 2007/46/EC

[10] EN 15722:2015, Intelligent transport systems - ESafety - ECall minimum set of data

[11] EN 16072:2015, Intelligent transport systems - ESafety - Pan-European eCall operating requirements

[12] EN 16454:2015, Intelligent transport systems - ESafety - ECall end to end conformance testing

[13] ETSI TS 101 539-2 Intelligent Transport System (ITS); V2X Applications; Intersection Collision Risk Warning (ICRW) application requirements specification

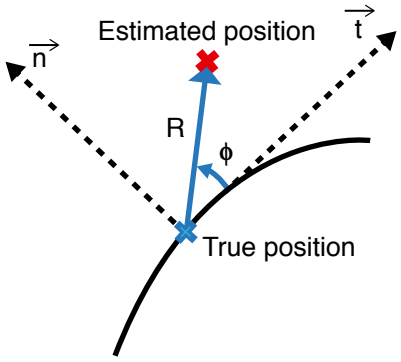
Appendix A: Example of a PVT error model

A.1 Principle of the model: Frenet frame and Laplace-Cauchy distributions

This appendix focuses on a positioning error model, proposed and assessed by the SaPPART COST Action, based on an experimental dynamic dataset acquired from a u-blox LEA5-T receiver in Frankfurt city centre.

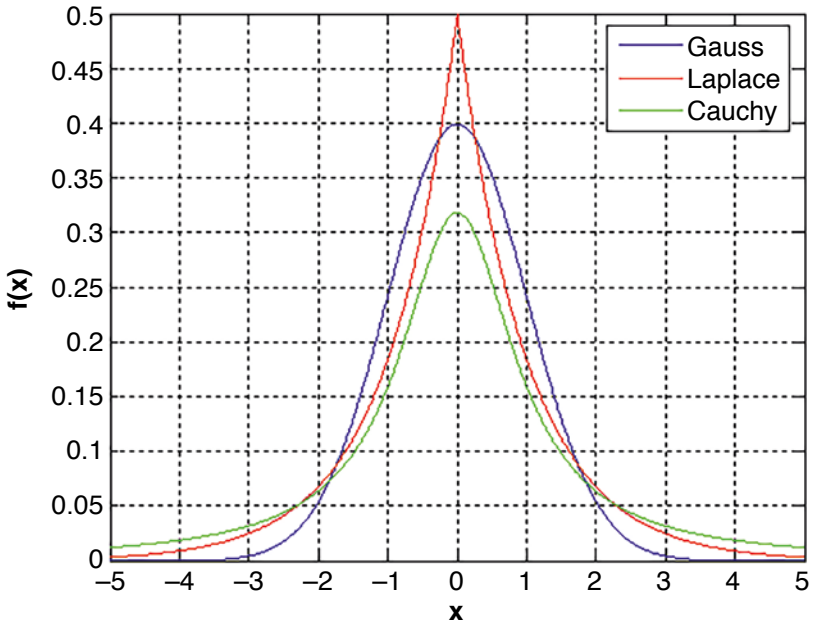
The model uses the local Frenet frame (see figure 24), tangent to the trajectory of the vehicle and the error vector is decomposed into its local polar components: radius and angle. This local frame, whose orientation is that of the trajectory, has interesting properties with regard to the error. This orientation is actually also that of the travelled street, where buildings create multipath or NLOS effects and the positioning error is known to be larger across-track than along-track.

Figure 24
The local Frenet frame



The proposed model is Laplace-Cauchy-based, meaning that the probability distributions that best fit the different components of the error are the Laplace distribution and the Cauchy distribution. The general shapes of these distributions, compared with the classical Gaussian distribution, are shown in figure 28. Their interesting property, as far as GNSS errors are concerned, is their “thick tail”, which is suited to the modelling of large errors.

Figure 25
The standard Gauss, Laplace and Cauchy distributions ($\mu = x_0 = 0$ and $\sigma = b = \gamma = 1$)



The Gauss, Laplace and Cauchy distributions are defined by their respective probability density function:

$$f_{Gauss}(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2} \quad (1)$$

$$f_{Laplace}(x) = \frac{1}{2b} e^{-\frac{|x-\mu|}{b}} \quad (2)$$

$$f_{Cauchy}(x) = \frac{1}{\pi\gamma} \frac{1}{\left(1+\frac{(x-x_0)^2}{\gamma^2}\right)} \quad (3)$$

In the local polar frame in which the errors are projected, the radius is modelled by piecewise constant values which are Laplace distributed, changing with probability p, on top of which is added a random walk from Cauchy distributed noise. The angle is modelled as a random walk from Cauchy distributed noise too. Truncations are also applied on the distributions, in particular to take into account the positive nature of the radius.

The probability p and the parameters of the statistical distributions are derived from experimental data.

Last, low-pass filters and normalization constants have to be designed.

The main idea which underlies this model is that stepwise errors will reflect urban perturbations due to buildings.

This Laplace-Cauchy-based model produced synthetic error distributions whose autocorrelations and CDFs are close to the true ones (obtained from experimental data),

and which generated error signals that look similar to the actual error signals from the u-blox receiver.

A.2 Expression of the model

A.2.1 Radius

Generating the error signal in the radius involves several steps. Firstly, a piecewise constant basis function made up of steps that are generated using a truncated Laplace distribution:

$$A'\left[n+1\right] = \frac{A\left[n\right] \text{ with probability } \left(1-p\right)}{L_r\left[n\right] \text{ with probability } p} \tag{4}$$

where *A'[n+1]* is the radius signal with no noise, *Lr[n]* is a realization of a truncated Laplace distribution giving the height of the new step, and p is the probability of a new step.

An additive noise, which is accumulated over each step, is then added:

$$A'[n + 1] = A[n + 1] + Cr[n] \tag{5}$$

where *Cr[n]* is a realization of a truncated Cauchy distribution.

To clarify, the Cauchy noise is not accumulated over the whole signal, but only over one step. It is reset when there is a new selection from the Laplace distribution.

The values of the parameters for generating *Lr[n]* and *Cr[n]* samples are estimated by means of Bayesian identification methods [8] assuming Gaussian distributions, with trial and error refinements. Finally, the signal is filtered using an auto-regressive filter (ARMAX type).

A.2.2 Angle

The angle is generated using a random walk given by:

$$f[n + 1] = f[n] + Sf[n] \tag{6}$$

where $S_\varphi[n]$ is a realization of a truncated Cauchy distribution.

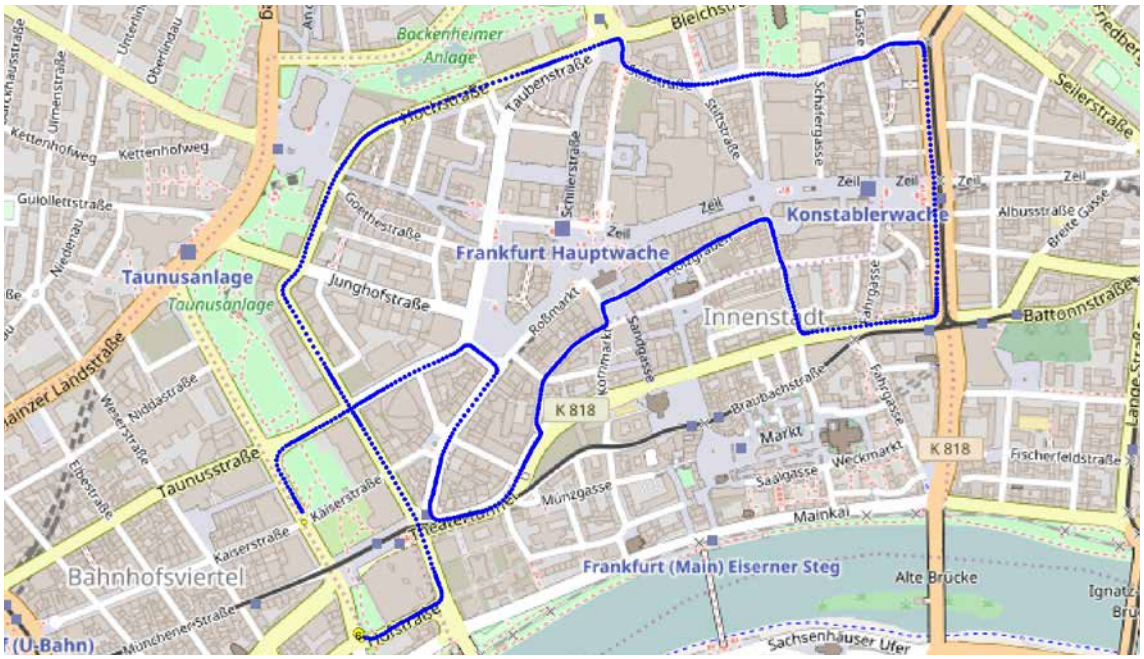
The signal is then low-pass filtered by a moving average filter. The values of the parameters needed to generate $S_\varphi[n]$ samples properly are determined experimentally.

A.3 Identification of the model parameters for the case of Frankfurt city centre

Experimental validation on a dataset acquired in Frankfurt, Germany, with a u-blox LEA5-T receiver, was conducted during the summer of 2015 by Q-Free and IFSTTAR [7]. It included the identification of the parameters, the simulation of error series, and their comparison to the true error. This dataset was provided by Q-Free.

The dataset was composed of 28 trajectories from Frankfurt, Germany, see figure 26. The 4.9km driving route passes through deep urban canyons with dense traffic, with an average driving speed of 15km/h, without tunnels or covered sections.

Figure 26
Driving route in Frankfurt recorded by the reference receiver

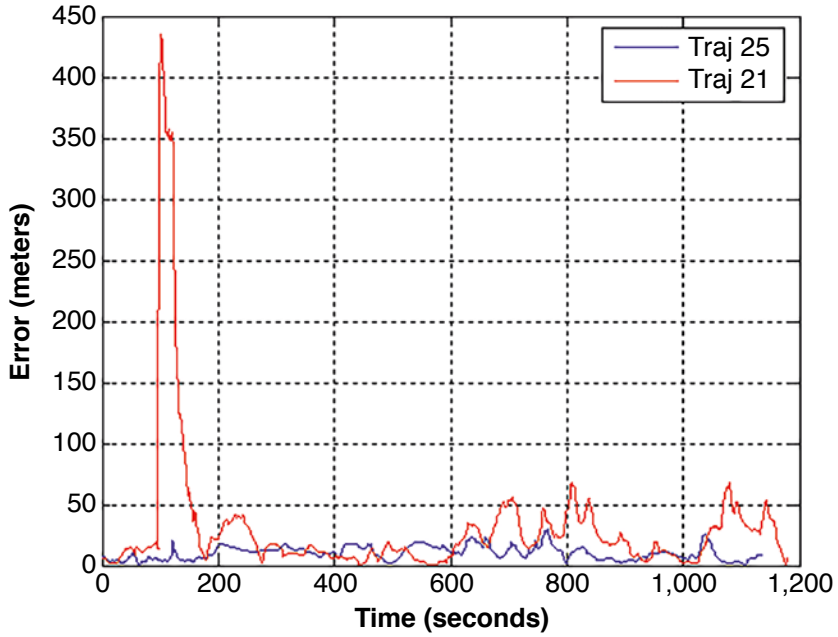


The reference track was provided by a Novatel CPOS combined with an Inertial Measurement Unit (IMU) by applying kinematic Precise Point Positioning (PPP) post processing using the Terrapos[5] software. It was accurate to within about one decimetre.

Manual inspection of the recorded trajectories revealed that 22 trajectories showed good GNSS performance from the u-blox receiver, 4 trajectories showed significantly larger errors, while 2 trajectories contained large error peaks of 400 m for approximately 30 seconds (see figure 27).

5. TerraPos is a software capable of accurate positioning based on post-processing of GPS-observations, without the use of reference stations or DGPS services, developed by the Norwegian company Terratec

Figure 27
Position error of the best (#25) and worst (#21) trajectories



The parameters of the model were estimated for all trajectories using the Hutter method [8].The Hutter method is a Bayesian exact regression of piecewise constant functions. It operates for Gaussian distributions of both the identified steps (their mean μ and standard deviation σ are identified) and the noise present above them (this noise is centred and its standard deviation σ_{noise} is identified as well).

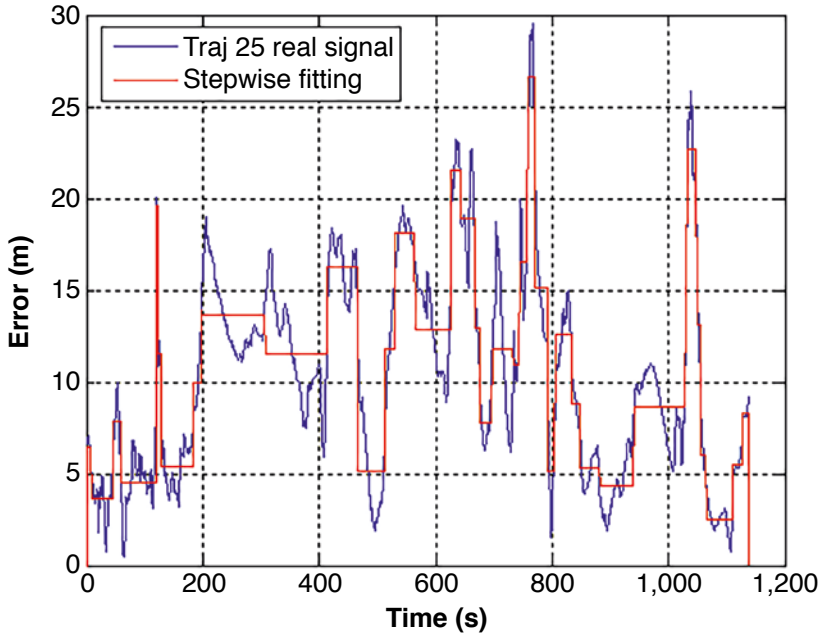
A similar optimization algorithm does not exist for Laplace distributed steps and, a fortiori, for the accumulated Cauchy noise. This is why, after running the Hutter algorithm and getting initial values for simulating a Laplace-Cauchy-based model, the set of parameters must be adjusted manually, by trial and error.

By analogy, the mean value μ from the Hutter method is kept the same as the initial mean value for the Laplace simulated signal. Similarly, the parameter b is fixed initially at σ. Since the Cauchy noise is accumulated in the simulation, its parameter γ must be much lower than the standard deviation σ_{noise} of the additive noise of the Hutter identification (otherwise the error would grow significantly within the duration of a step), and this is typically to be fixed in simulation by trial and error.

When simulating, the analysis of the signal produced is compared visually to the original signal time series. The CDFs are also compared, as are the autocorrelations.

Despite the fact that the Hutter method is exact, the complete process leading to the final parameters is an approximation, with empirical results. Trajectory 25, the best in terms of error, was chosen to estimate an outline of the complete set of parameters. Figure 31 shows the result of a Hutter-derived estimation of step probability where the stepwise red line is the approximation.

Figure 28
Hutter estimation of stepwise approximation and step probability



The filter applied to the radius has been modelled by an autoregressive moving average with exogenous inputs (ARMAX) filter, and identified with the stepwise approximation as the signal input, and the original error as the signal output. The number of poles (here: 2) and zeroes (here: 0) must be determined by trial and error. Moreover, a delay of 1 sample (1 second) was applied and the noise disturbance was pink (using 1 previous value). To summarize, the *armax* Matlab function was called with [na=2, nb=1, nc=1, nk=1] parameters. The trial and error approach is judged successful when the residuals of the ARMAX identification are uncorrelated. This can be checked with the *resid* Matlab function.

The filter applied to the angle here is empirically a moving average of the 10 previous input values.

Table 12 shows the obtained parameters.

Table 12
Simulation parameters

Parameter	Selected value
Laplace (radius)	
μ	11 m
b	36 m
min	0.1 m
max	22 m
Cauchy (radius)	
x_0	0
γ	0.5 m
min	-1.0 m
max	1.0 m
Filter coef (radius) a	{ 1, -1.728, 0.7544 }
Filter coef (radius) b	0.02518
Step probability p (radius)	0.032
Cauchy (angle)	
x_0	0
γ	0.0161 rad
min	-pi
max	pi
Filter coef (angle) a	1
Filter coef (angle) b	{ 0.1, 0.1, 0.1, 0.1, 0.1, 0.1, 0.1, 0.1, 0.1, 0.1 }

Radius errors were simulated using the parameters selected within the model. One simulation result is shown in figure 32 along trajectory 25. The autocorrelation can be seen in figure 33. It is somewhat lacking in the second lobes, but the simulation is quite good as can be seen in figure 34 where the CDFs of the simulated signal and the targeted signal are close together.

Figure 29
Change over time of of simulated and target signals for trajectory 25

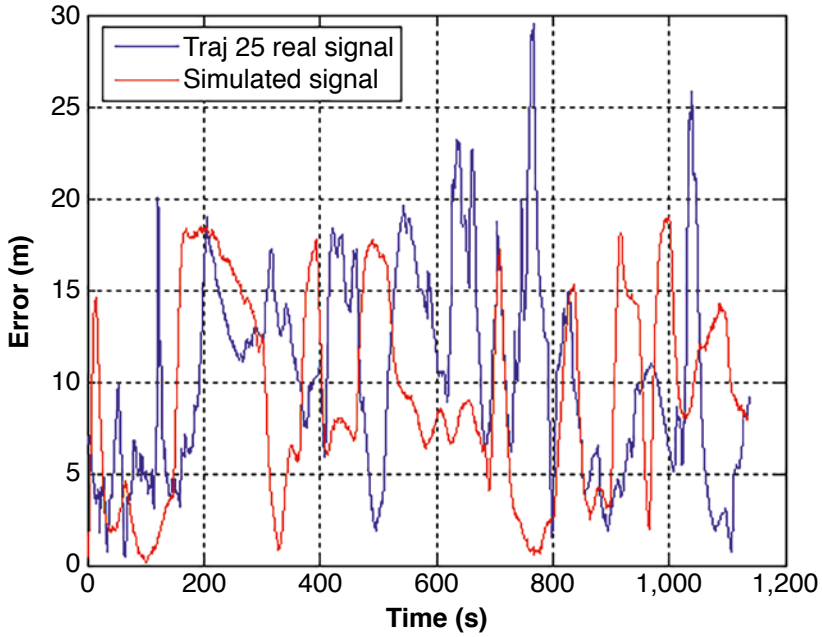


Figure 30
Autocorrelation function of signals for trajectory 25

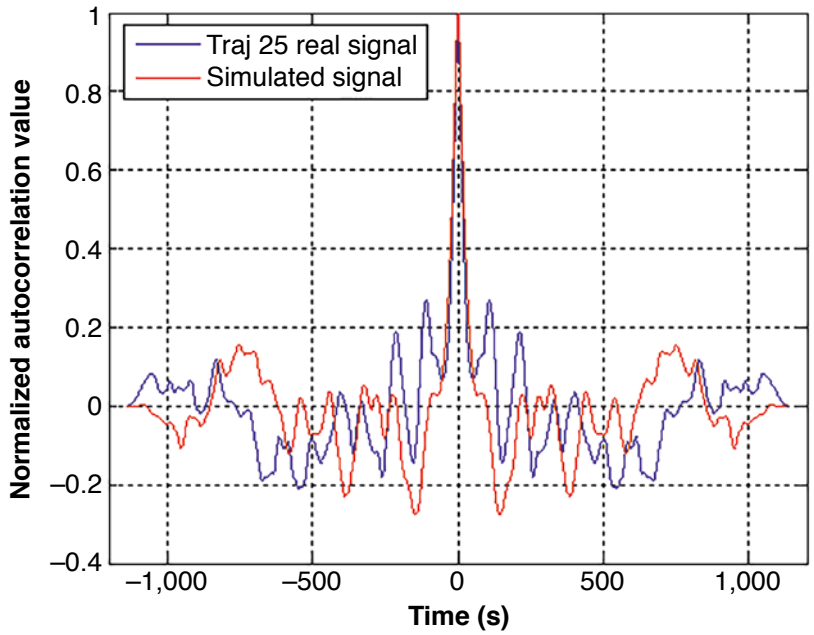
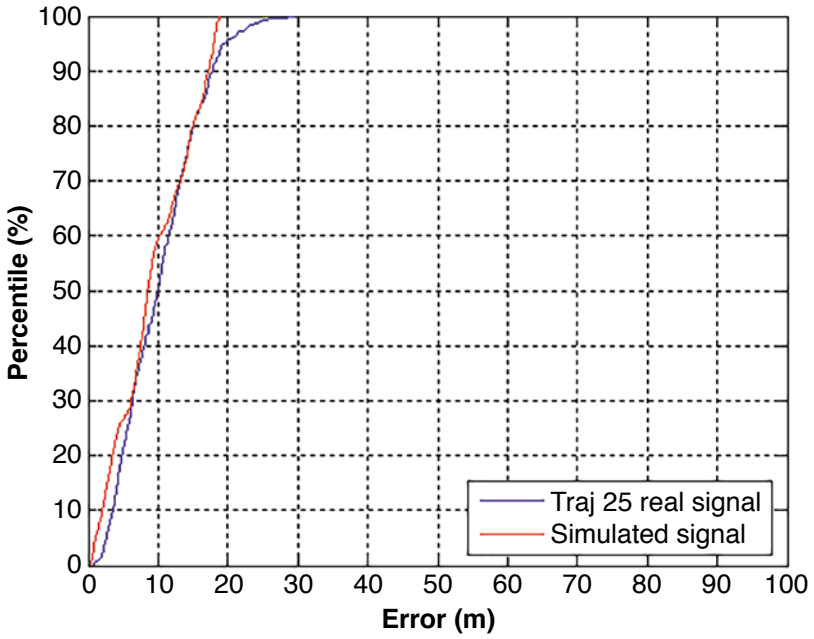


Figure 31
Cumulative distributions of signals for trajectory 25



A.4 Limits of the model

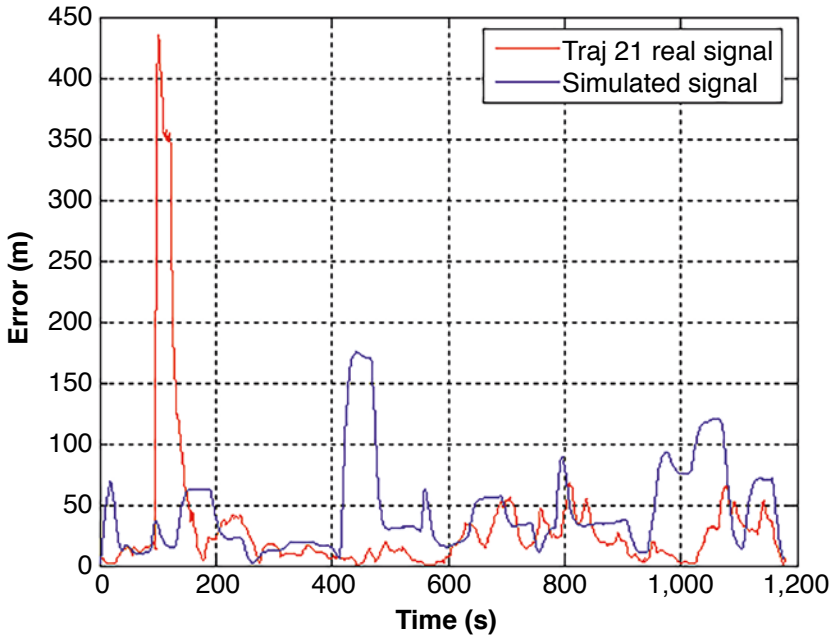
Figure 32 shows an attempt to reproduce trajectory 21 which exhibited a large peak. The considerable error observed around t = 100 s is very sudden, but it lasts for some epochs. The lasting nature of the error is probably due to filtering in the GNSS receiver.

For this purpose, the upper bound of the truncated Laplace distribution has been moved from 22 m to 400 m, all other parameters being unchanged, to try to reproduce very high peaks.

As can be observed in figure 32, the simulated error signal is not satisfactory. The correlation is not a good match and neither is the CDF (not shown). There is obviously a model mismatch: the large transient peaks of trajectory 21 cannot be properly reproduced by our model, even after increasing the upper bound of the truncated Laplace distribution. More parameters have to be re-identified.

This highlights the limits of the approach: the model identified from a unique seed trajectory (in our case, trajectory 25) cannot simulate trajectories that are exceptional and too distant from the seed.

Figure 32
Time evolution of simulated and target signals for trajectory 21



A.5 Conclusions

It has been demonstrated that the methodology consisting of collecting GNSS solutions, comparing these solutions to ground truth and elaborating a model of PVT errors in order to generate replicas for simulation purpose is feasible. This feasibility has been demonstrated in the case of a complex scenario, in a challenging urban GNSS environment with typically urban driving patterns at low speed and with frequent stops and starts.

This model is based upon the basic idea that noisy stepwise errors correctly reflect urban perturbations due to buildings. This stepwise structure was identified thanks to the Hutter method and adapted to the specificity of the GNSS errors by switching the probability laws used in the Hutter method from Gauss to Laplace and Cauchy.

The Laplace-Cauchy model that was thus identified, with parameters estimated from one unique trajectory, was representative of most trajectories, all that was required being the tuning of the upper truncation bound of the Laplace distribution. Only exceptional trajectories with high peaks could not be correctly simulated. Hence, special care should be taken when choosing the seed trajectory and the limits of the synthetic trajectories that can be generated from this unique seed should be taken into account.

Naturally, what has been achieved in SaPPART, in terms of modelling results, is dependent on a receiver model used under specific conditions. However, the methodology itself can be re-used in another context and for any other GNSS-based positioning equipment (i.e. the methodology is transferable).

List of acronyms

ADAS	Advanced Driving Assistance System
CCR	Correct Charging Rate
CDF	Cumulative Distribution Function
CEN	Comité Européen de Normalisation/European Committee for Standardization
CENELEC	Comité Européen de Normalisation en Electronique et Electrotechnique/ European Committee for Electrotechnical Standardization
CEP	Circular Error Probable
COST	European COoperation in Science and Technology
DOP	Dilution Of Precision
E2E	End-to-End
eCall	Emergency Call
GBPT	GNSS-based positioning terminal
GLONASS	Global'naya Navigatsionnaya Sputknikkovaya Sistema
GNSS	Global Navigation Satellite System
GPS	Global Positioning System
HMI	Human Machine Interface
HPE	Horizontal Position Error
ICRW	Intersection Collision Risk Warning
IMU	Inertial Measurement Unit
IR	Integrity Risk
ISA	Intelligent Speed Adaptation
ITS	Intelligent Transport Systems
KPI	Key Performance Indicator
LE	Longitudinal Error
MEO	Medium Earth Orbit
MI	Misleading Information
MSD	Minimum Set of Data (eCall)
NLOS	Non-Line-Of-Sight
OCR	Over Charging Rate
PDF	Probability Density Function

PL	Protection Level
PPP	Precise Point Positioning
PSAP	Public-Safety Answering Point (eCall)
PVT	Position, Velocity and Time
RUC	Road User Charging
SD	Standard Deviation
SNR	Signal-to-Noise Ratio
TIR	Target Integrity Risk
TTFF	time-to-first-fix
VG	Virtual Gantries
WGS 84	World Geodetic System 1984

List of figures

List of tables

Publication data form

Collection techniques and methods		
ISSN 2492-5438	**ISBN** Print 978-2-85782-727-6 PDF 978-2-85782-728-3	**Ref.** TMI2
Title SaPPART Handbook		
Subtitle Assessment of positioning performance in ITS applications		
Author COST Action TU1302		
Sponsor name and address COST Association avenue Louise 149 1050 Brussels Belgium		
Publication date June 2017	**Language** English	

Summary

This handbook is the second deliverable of SaPPART COST Action, dedicated to performance issues, when positioning performance is essential to the fulfilment of the requirements of the whole ITS system. It starts by illustrating the non-straightforward nature of the role of positioning information in some emblematic applications and introduces a simulation method sensitivity analysis, as a tool to make the right choice of positioning terminal for a given application. Then, the Handbook discusses the error sources at the terminal level and introduces a model of the horizontal position error in an urban environment. In the final part, this error model and the sensitivity analysis are applied to two examples of ITS systems, namely Road User Charging and eCall, in order to illustrate how sensitive these systems are to the positioning performance.

Key words positioning, GNSS, intelligent transport systems, performance, error model	
Number of pages 77	**Price** free of charge

Fiche bibliographique

Collection techniques et méthodes		
ISSN 2492-5438	**ISBN** Papier 978-2-85782-727-6 PDF 978-2-85782-728-3	**Réf.** TMI2
Titre Manuel SaPPART		
Sous-titre Évaluation des performances de positionnement dans les applications STI		
Auteur COST Action TU1302		
Nom et adresse du financeur COST Association avenue Louise 149 1050 Brussels Belgium		
Date de publication Juin 2017	**Langue** anglais	

Résumé

Ce manuel est le deuxième livrable de l'Action COST SaPPART, dédié aux problèmes de performances, quand les performances de positionnement sont essentielles pour satisfaire les besoins du système STI dans son ensemble. D'abord, il illustre la complexité du rôle de l'information de position dans quelques applications emblématiques et introduit une méthode de simulation appelée analyse de sensibilité, en tant qu'outil d'aide au choix du terminal de positionnement pour une application donnée. Ensuite, le manuel discute les sources d'erreur au niveau du terminal et introduit un modèle de l'erreur de positionnement horizontale dans un environnement urbain. Dans la dernière partie, ce modèle d'erreur et l'analyse de sensibilité sont appliqués à deux exemples de systèmes STI, le télépéage et l'eCall, afin d'illustrer la sensibilité de ces systèmes à la performance de positionnement.

Mots clés localisation, GNSS, systèmes de transport intelligents, performance, modèle d'erreur	
Nombre de pages 77	**Prix** gratuit

Published by Ifsttar - Legal submission: June 2017
ISBN: 978-2-85782-728-3 - ISSN: 2492-5438
Photo credits: EPFL – Pierre-Yves Gilliéron / ©ESA–J. Huart
Graphic design and layout by Ifsttar and STDI

Ifsttar Head Office: 14-20 boulevard Newton – Cité Descartes – Champs-sur-Marne – 77447 Marne-la-Vallée cedex 2
www.ifsttar.fr

This handbook is the second deliverable of SaPPART COST Action, dedicated to performance issues, when positioning performance is essential to the fulfilment of the requirements of the whole ITS system. It starts by illustrating the non-straightforward nature of the role of positioning information in some emblematic applications and introduces a simulation method sensitivity analysis, as a tool to make the right choice of positioning terminal for a given application. Then, the handbook discusses the error sources at the terminal level and introduces a model of the horizontal position error in an urban environment. In the final part, this error model and the sensitivity analysis are applied to two examples of ITS systems, namely Road User Charging and eCall, in order to illustrate how sensitive these systems are to the positioning performance.

Ce manuel est le deuxième livrable de l'Action COST SaPPART, dédié aux problèmes de performances, quand les performances de positionnement sont essentielles pour satisfaire les besoins du système STI dans son ensemble. D'abord, il illustre la complexité du rôle de l'information de position dans quelques applications emblématiques et introduit une méthode de simulation appelée analyse de sensibilité, en tant qu'outil d'aide au choix du terminal de positionnement pour une application donnée. Ensuite, le manuel discute les sources d'erreur au niveau du terminal et introduit un modèle de l'erreur de positionnement horizontale dans un environnement urbain. Dans la dernière partie, ce modèle d'erreur et l'analyse de sensibilité sont appliqués à deux exemples de systèmes STI, le télépéage et l'eCall, afin d'illustrer la sensibilité de ces systèmes à la performance de positionnement.

COST is supported by the EU Framework Programme Horizon 2020

IFSTTAR

LES COLLECTIONS DE L'IFSTTAR

ISSN: 2492-5438
Ref: TMI 2
Photo credit:
Pierre-Yves Gilliéron - EPFL
June 2017