

ASSESSMENT OF SOFTWARE-DEFINED GNSS RECEIVERS

Carles Fernández-Prades, Javier Arribas, and Pau Closas

Centre Tecnològic de Telecomunicacions de Catalunya (CTTC), 08860, Barcelona, Spain
Statistical Inference for Communications and Positioning Department
Email: {cfernandez, jarribas, pclosas}@cttc.cat

ABSTRACT

The aim of this paper is to trigger a conversation about the assessment, definition of metrics and testing procedures of software-defined GNSS receivers. While the evaluation of *traditional* (i.e., built on application-specific integrated circuit technology) GNSS receivers is now well-understood, and enjoys both a solid testing industry providing the required equipment and universally agreed figures of merit, the particularities of software-defined radio technologies claim for a more comprehensive approach. In order to account for such a multi-faceted nature, the authors identify sixteen design forces, or dimensions in which a software-defined GNSS can improve. Upon those definitions, a wide list of performance indicators, metrics and procedures are then proposed for each of the identified thrusts. The list can be used as a generative source of ideas when defining key performance indicators in projects, products or services involving a software-defined GNSS receiver.

Index Terms— Performance analysis, Satellite navigation systems, Receivers, Software defined radio.

1. INTRODUCTION

A GNSS receiver is a complex device which performance is affected by a wide range of internal and external factors. To the best of the authors' knowledge, the first formal effort to define testing procedures for GPS receivers is found in [1], a work that anticipated the key concepts of the Standard 101 published by the Institute of Navigation in 1997 [2]. Such procedures have been widely accepted by the GNSS industry and, two decades later, world-class testing firms are still proposing them in their white papers (see Agilent Technologies [3], National Instruments [4], Rohde & Schwarz [5], or Spirent [6]). In summary, the set of proposed tests measure

receiver's sensitivity in acquisition and tracking, diverse time-to-first-fix and reacquisition times, static and dynamic location accuracy and robustness to multipath and radio frequency (RF) interferences.

The very nature of software-defined radio technology requires a broader approach. A GNSS receiver in which the baseband processing chain is implemented in software and executed by a general-purpose processor in a computer system has other design forces equally important and clue for real impact and to reach technical, market and social success, but they are usually not captured by *traditional* GNSS testing procedures.

Next section identifies sixteen dimensions in which the performance and features of a software-defined GNSS receiver can be evaluated. This taxonomy allows comparison of different implementations.

2. DESIGN FORCES

The design of a GNSS software-defined receiver needs to resolve some design forces that could appear as antithetical, (e.g., portability vs. efficiency, openness vs. marketable product), and a "sweet spot" must be identified according to the targeted user and application. This section provides the definition of design forces to be considered when planning the assessment of a software-defined GNSS receiver. Such definitions, although they are put in the context at hand, are kept general (most of them directly extracted from Wikipedia) in order to capture the concepts in their widest sense. They are not pretended to be orthogonal.

Accuracy: In this context, it refers to how close a Position-Velocity-Time (PVT) solution is to the true (actual) position (that is, a measure of the bias or systematic error). Its measurement requires a reference (fiducial) position in the case of static positioning, and a controlled mobile platform for dynamic positioning.

Availability: The degree to which a system, subsystem or equipment is in a specified operable and committable state at the start of a mission, when the mission is called for at an unknown, random time. Simply put, availability is the proportion of time the software receiver is in a functioning condition.

This work has been developed in the frame of AUDITOR. This project has received funding from the European GNSS Agency under the European Unions Horizon 2020 research and innovation programme under grant agreement no. 687367, as well as by the Spanish Ministry of Economy and Competitiveness and the European Regional Development Fund, through project TEC2015-69868-C2-2-R (MINECO/FEDER) and by the Government of Catalonia under Grant 2014-SGR-1567.

Efficiency: In this context, it refers to optimizing the speed and memory requirements of the software receiver. Specifically, we are interested on how fast the software receiver can process the incoming signal, and in particular if signal processing up to the position fix can be done in real-time using a RF front-end (and how many channels it can sustain in parallel). Efficiency can also refer to the optimization of power consumption required by the processor running the software receiver.

Flexibility: In the context of engineering design, it refers to the ability of a system to respond to potential internal or external changes affecting its value delivery, in a timely and cost-effective manner.

Interoperability: It refers to the ability of making systems work together. In particular, the possibility to exchange information with other free and proprietary software, devices and systems, including GNSS signals, RF front-ends, external assistance, and all sort of information-displaying or sensor data fusion applications via standard outputs.

Maintainability: It refers to the ease with which a product can be maintained in order to isolate and correct defects or their cause, repair or replace faulty or worn-out components without having to replace still working parts, prevent unexpected breakdowns, maximize a product's useful life, maximize efficiency, reliability, and safety, meet new requirements, make future maintenance easier, or cope with a changed environment.

Marketability: A measure of the ability of a security to be bought and sold. If there is an active marketplace for a security, it has good marketability. Producing higher quality products and pricing them competitively can increase marketability, attracting consumers wanting to choose our product over an equally priced item with less quality. But marketability can be also increased by radically changing the features of existing products by means of a technology shift, attracting customers with new product/service benefits (lower prices, openness, usefulness, sense of community, closer interaction between users and the actors of the value chain) while approaching quality of well-established products on the market, as well as opening blue oceans of market space.

Portability: It refers to the usability of the same software in different environments.

Popularity: It is a complex social phenomenon with no agreed upon definition. It can be defined in terms of liking, attraction, dominance, superiority, or just being *trendy*. Through peer influence, target objects can quickly skyrocket in how pervasive they are in the user community and because popularity is judged in a social context, the more pervasive it is, the more popular it might be considered.

Reliability: It describes the ability of a system or component to function under stated conditions for a specified period of time. Reliability refers to the consistency of the results provided by a system; internal and external reliability are, respectively, the ability to detect gross errors and the effect of

an undetected blunder on the solution.

Repeatability: It is related to the spread of a measure, also referred to as precision. It refers to how close a position solution is to the mean of all the obtained solutions, in a static location scenario.

Reproducibility: It refers to the ability of an entire experiment or study to be reproduced, either by the researcher or by someone else working independently. It is one of the main principles of the scientific method and relies on *ceteris paribus* (other things being equal). When applied to software engineering, it has other additional implications such as in security (*i.e.*, gaining confidence that a distributed binary code is indeed coming from a given verified source code).

Scalability: It refers to the ability of the software to handle a growing amount of work in a capable manner, or its ability to be enlarged to accommodate that growth.

Testability: When referred to software, it is the degree to which a software artifact (*i.e.* a software system, software module, requirements, or design document) supports testing in a given test context. Testability is not an intrinsic property of a software artifact and cannot be measured directly (such as software size). Instead, testability is an extrinsic property that results from interdependency of the software to be tested and the test goals, test methods used, and test resources (*i.e.*, the test context). Testability can be understood as visibility and control. Visibility is our ability to observe the states, outputs, resource usage and other side effects of the software under test. Control is our ability to apply inputs to the software under test or place it in specified states.

Openness: It is a relative characteristic that refers to the degree to which something is accessible to view, modify and use. From a social perspective, openness is a core characteristic of an infrastructure that conveys and reinforces sharing, reciprocity, collaboration, tolerance, equality, justice and freedom. Understanding this *megatrend* (as defined in [7]) and its rolling effects can provide valuable information for developing futuristic scenarios and can subsequently help to shape current actions in anticipation of that future.

Usability: It refers to the degree to which a software can be used by specified consumers to achieve quantified objectives with effectiveness, efficiency, and satisfaction in a quantified context of use.

Next section is devoted to the definition of indicators (procedures, metrics and check points) for each of the identified design forces.

3. KEY PERFORMANCE INDICATORS

Key Performance Indicators (KPIs) are goals or targets that measure how well a given activity is doing on achieving its overall operational objectives or critical success factors. KPIs must be objectively defined in order to provide a quantifiable

and measurable indication of the product or service development progress towards achieving its goals.

S.M.A.R.T. is an acronym mentioned for the first time in 1981 [8], and it is usually referred to when identifying and defining KPIs, in order to remind their desirable features:

- Specific: Is this KPI too broad, or is it clearly defined and identified?
- Measurable: Can the measure be easily quantified?
- Attainable: Is it realistic for us to obtain this measure within the given project framework? Can we take the appropriate measures to implement this KPI and see changes?
- Realistic: Is our measure practical and pragmatic?
- Timely: How often are we going to be able to look at data for its measure?

Hence, KPIs are not universal but based on the very single project, product or service in which they are going to be applied. This section suggests a wide list of indicators, derived from the design forces defined in Section 2, to be considered when assessing the quality of a software-defined GNSS receiver. Its degree of *S.M.A.R.T.-ness* in every particular context may vary.

3.1. Indicators of Accuracy

Upon the definition of the GNSS satellite coordinate reference system (expressed as “ITRFyy at epoch yyyy.y”, see [9]) and ellipsoid (*e.g.*, WGS 84); the local geographic coordinate reference system (providing transformation parameters, if applicable) and ellipsoid; and, in case of differential GNSS configurations, the datum of the differential source, possible accuracy indicators are:

- Stand-alone static position accuracy.
 - Position accuracy results are given in meters of error with respect to a reference (fiducial) point previously measured in a geodetic survey, or defined by the testing equipment (see Section 3.2.1). Two of the most commonly used confidence measurements for 2D positioning are the Distance Root Mean Square (DRMS) and the Circular Error Probability (CEP); and the Mean Radial Spherical Error (MRSE) and the Spherical Error Probable (SEP) when measures are expressed in 3D. See definitions in Table 1, where standard deviations are computed as

$$\sigma_E^{(accuracy)} = \sqrt{\frac{1}{L-1} \sum_{l=1}^L (E[l] - E_{ref})^2} \quad (1)$$

where E_{ref} is the East coordinate of the reference location. Similar expressions can be defined for the North and Up coordinates.

Measure	Formula	Confidence region probability
2D DRMS	$\sqrt{\sigma_E^2 + \sigma_N^2}$	65%
2D 2DRMS	$2\sqrt{\sigma_E^2 + \sigma_N^2}$	95%
2D CEP	$0.62\sigma_N + 0.56\sigma_E$, if $\frac{\sigma_N}{\sigma_E} > 0.3$	50%
3D MRSE	$\sqrt{\sigma_E^2 + \sigma_N^2 + \sigma_U^2}$	61%
3D SEP	$0.51(\sigma_E^2 + \sigma_N^2 + \sigma_U^2)$	50%

Table 1. Most common positioning error measures, where σ_E^2 , σ_N^2 , and σ_U^2 are the error variance in a local East-North-Up (ENU) coordinate reference system, respectively [10].

- Stand-alone dynamic position accuracy.
 - In this case, the reference is not a single point but a timed trajectory. Different trajectories and locations can be averaged to mitigate differences due to satellite visibility and geometry. Same metrics than for static positioning, where the position references will now have a time index.
- Differential GNSS static and dynamic accuracies.
 - Same metrics than in stand-alone configurations.

3.2. Indicators of Availability

Possible availability indicators are:

- Proportion of time the software receiver is up and running in a continuous manner.
- Acquisition sensitivity for each targeted GNSS signal, in dBm (see Section 3.2.2).
- Tracking sensitivity for each targeted GNSS signal, in dBm (see Section 3.2.3).
- Time to First Fix, in seconds (see Section 3.2.4), for the following scenarios:
 - Cold start, defined with the following assumptions:
 - * Time is unknown.
 - * Current almanac and ephemeris unknown.
 - * Position unknown.
 - Warm start, defined with the following assumptions:
 - * Time is known.
 - * Current almanac is known.
 - * No ephemeris (or the data is more than four hours old).
 - * Position within 100 km of last fix.
 - Hot start, defined with the following assumptions:
 - * Time is known.
 - * Current almanac is known.

- * Current ephemeris are known.
- * Position within 100 km of last fix.

- Reacquisition time, in seconds (see Section 3.2.5), per targeted GNSS signal.

In case of using differential GNSS techniques:

- Availability and continuity of a minimum number of input datastreams.
- Availability of corrections for precise positioning.
- Corrections latency / generation time.
- Convergence time to subdecimeter level.
- Phase ambiguity fixing success rate.
- Baseline maximum length.

In case of using assisted GNSS techniques:

- Availability of an external service delivering assisted GNSS data.

3.2.1. Generation of testing inputs

Most of the testing procedures described below and in Section 3.1 require advanced RF GNSS signal generators in order to reproduce the radioelectric environment that a GNSS receiver antenna would sense in a variety of controlled scenarios. Usual features of such equipments include the number and type of concurrently generated GNSS signals, bands and channels, the possibility to define custom timed position positions and trajectories, and the simulation of effects as atmospheric errors, satellite clock and ephemeris errors, receiver and satellite motion, obscuration and multipath, receiver antenna characteristics and presence of interferences. In addition, software-defined receivers may be able to operate by reading GNSS data from files or from a network-delivered data stream instead of from the output of a RF front-end. Hence, in some contexts, RF GNSS signal generators can be replaced or complemented by software-generated data sets and/or actual signals broadcast by GNSS constellations.

3.2.2. Measuring acquisition sensitivity

Acquisition sensitivity determines the minimum signal power threshold that allows the receiver to successfully perform a cold start TTFF within a specified time frame. The generation of testing inputs is as follows: fixing the number of visible satellites to one, the power level of the received signal is set such that the GNSS software receiver under test can detect the single GNSS satellite signal within a given probability of detection. The power level of the GPS satellite signal is then decreased until the GNSS receiver is not able to acquire that satellite signal. This power level and the corresponding GNSS software receiver under test carrier-to-noise density ratio (C/N_0) should be collected as data. The received power level at the beginning of this scenario is -140 dBm, and it is decreased by 1 dB in each acquisition procedure.

3.2.3. Measuring tracking sensitivity

Tracking sensitivity refers to the minimum signal level that allows the receiver to maintain a location fix within some specified degree of accuracy. The generation of testing inputs is as follows: fixing the number of visible satellites to one, the power level of the received signal is set such that the GNSS software receiver under test can identify the single GNSS satellite signal. The power level of the GNSS satellite signal is then decreased until the GNSS receiver loses tracking of the single satellite. This power level and the corresponding GNSS receiver C/N_0 should be collected as data. The received power level at the beginning of this scenario is -130 dBm, and it is decreased by 1 dB at 60-second intervals. Another receiver sensitivity test is to measure the power level and C/N_0 level at which 3D location fix is lost. This is a similar procedure as above but using eight visible satellites. The power level of the received signals are then decreased until the 3D location fix is lost. Again the power level and the corresponding GNSS software receiver under test C/N_0 are collected as data.

3.2.4. Measuring Time to First Fix

Before taking each TTFF measurement, the receiver must be set in the states defined in Section 3.1, referred to as *cold*, *warm* and *hot* starts. In order to ensure meaningful statistical results, the Institute of Navigation suggests collecting a minimum of 20 TTFF measurements [2]. Other authors (see [11]) suggest a duration of 8 hours.

3.2.5. Measuring reacquisition time

Reacquisition time characterizes the performance of the receiver in a scenario where the signal is greatly reduced or interrupted for some short period of time and is then restored. An example of this would be a vehicle going through a tunnel or under some heavy tree cover. In this case the receiver is briefly unable to track most or all of the satellites, but must re-acquire (track) the signal when “visibility” is restored. The generation of testing inputs is as follows: a static scenario is generated for a 32-minute period of time. Following an initial 300-second interval of full satellite visibility to allow a successful cold start, all simulated satellite signals are switched off for 5 seconds every 30 seconds. Reacquisition times for individual satellites and for the complete navigation solution can be obtained by comparing the receivers logged data with the satellite on/off times in the test scenario.

3.3. Indicators of Efficiency

Possible efficiency indicators are:

- Number of parallel channels that the software receiver can sustain in real time, given the targeted signal(s) (GPS L1 C/A, Galileo E1B, etc.) of each channel, the

sampling rate, the sample data format and the computational resources available for signal processing.

- Power consumption (in watts) for a given computing platform executing the software receiver and a given computational load in terms of number of signals and channels to be processed. Power consumption sometimes is given as current (in mA) for a given fixed voltage (in volts).
- Availability of profiling tools for identifying processing bottlenecks and measuring computational performance in the supported processing environments (processor architecture, operating system, etc.).

3.4. Indicators of Flexibility

Possible flexibility indicators are:

- Possibility to either use synthetically generated or real-life GNSS signals.
- Possibility to process signals either in real time or in post-processing time (only limited by the computational capacity of the processing platform executing the software receiver).
- Possibility to use different RF front-ends.
- Possibility to define custom receiver architectures.
- Possibility to easily define / interchange implementations and parameters for each processing block.
- Possibility to change parameters while the software is executing.
- Availability of *operation modes*, as combinations of:
 - Single / multiple frequency bands.
 - Single / multiple constellations.
 - Stand-alone / assisted / differential GNSS.

3.5. Indicators of Interoperability

Possible interoperability indicators are:

- Number of GNSS signals, defined as combinations of frequency band and channel or code, from which GNSS observables (*i.e.*, measurements of pseudorange, carrier phase, Doppler and signal strength) can be generated (see [12] for definitions):
 - GPS
 - * L1 band: 1575.42 MHz
 - C/A; L1C (D); L1C (P); L1C (D+P); P (AS off); Z-tracking and similar (AS on); Y; M; and codeless.
 - * L2 band: 1227.60 MHz
 - C/A; L1(C/A)+(P2-P1) (semi-codeless); L2C (M); L2C (L); L2C (M+L); P (AS off); Z-tracking and similar (AS on); Y; M; and codeless.
 - * L5 band: 1176.45 MHz

· I; Q; I+Q.

– Galileo

- * E1 band: 1575.42 MHz
 - A PRS; B I/NAV OS/CS/SoL; C no data; C+B; A+B+C.
- * E5a band: 1176.45 MHz
 - I F/NAV OS; Q no data; I+Q.
- * E5b band: 1207.14 MHz
 - I I/NAV OS/CS/SoL; Q no data; I+Q.
- * E5 (E5a+E5b) band: 1191.795 MHz
 - I; Q; I+Q.
- * E6 band: 1278.75 MHz
 - A PRS; B C/NAV CS; C no data; B+C; A+B+C.

– GLONASS

- * G1 band: $1602 + k \cdot 9/16$ MHz, $k = -7, \dots, +12$.
 - C/A; P.
- * G2 band: $1246 + k \cdot 716$ MHz
 - C/A (GLONASS M); P.
- * G3 band: 1202.025 MHz
 - I; Q; I+Q.

– Beidou

- * B1 band: 1561.098 MHz
 - I; Q; I+Q.
- * B2 band: 1207.14 MHz
 - I; Q; I+Q.
- * B3 band: 1268.52 MHz
 - I; Q; I+Q.

– Depending of the region of use, other satellite-based signals can be available:

- * SBAS: C/A in the L1 band; I, Q, I+Q in L5.
- * QZSS: C/A, L1C (D), L1C (P), L1C (D+P), and L1-SAIF in the L1 band; L2C (M), L2C (L) and L2C (M+L) in the L2 band; I, Q, and I+Q in the L5 band; and S, L, and S+L in the LEX(6) band located at 1278.75 MHz.
- * IRNSS: A SPS, B RS (D), C RS (P), and B+C in the L5 band; A SPS, B RS (D), C RS (P), and B+C in the S band, located at 2492.028 MHz.

• For RF front-end(s):

- Availability of software drivers.
- For antenna(s):
 - * Antenna identification number and type. In case of multiple antennas: geometrical arrangement.
 - * Average antenna phase center relative to the antenna reference point (ARP) for each specific frequency and satellite system.

- * Orientation of the antenna zero-direction as well as the direction of its vertical axis (bore-sight), if mounted tilted on a fixed station, or XYZ vector in a body-fixed system, in case of mounted on a moving platform. All units in meters.
- * If the antenna is physically apart from the front-end: cable category and length, connectors type.
- Sampling frequency, center frequency and intermediate frequency (per frequency band).
- Availability of external clock input.
- Sample bit length and interpretation (baseband complex samples or interleaved I&Q samples at a given intermediate frequency, inverted spectrum indicator).
- If the RF front-end is used as a remote radio-head in a cloud-based system:
 - * Throughput of the link between the remote radio-head and the processing system.
- For raw GNSS (and possibly other sensors) data stored digitally, support of the fundamental data collection topologies, as defined by the ION GNSS SDR Standard Working Group (see [13]):
 - Single band, single-stream, single file.
 - Multi-band, single-stream, single file.
 - Multi-stream, single file.
 - Multi-sensor, single file.
 - Temporal splitting of files.
 - Spatial splitting of files.
 - Spatial-temporal splitting.
- Support of sample formats for the exchange of raw GNSS data (see [13]):
 - Quantization: 1, 2, 4, 8, 16, 32 or 64 bits per sample.
 - Encoding: sign, sign-magnitude, signed integer, offset binary or floating point.
- Support of output formats:
 - Type and frequency of real-time generated RTCM messages (defined in [14]), streamed over a communication network as defined by the Networked Transport of RTCM via Internet Protocol (NTRIP, see [15]). Specify RTCM version.
 - RINEX observation and navigation data files (see [12]). Specify version: 2.10, 2.11, 3.02, 3.03.
 - GIS formats: KML, GeoJSON, SHP.
 - Application-specific messages (*e.g.*, NMEA 0183 / 2000, ISOBUS, proprietary/custom).
- Support of data link protocols:
 - Ethernet (IEEE 802.3ab / 802.3ae / others).
 - Wireless LAN (IEEE 802.11 family).

- Bluetooth (specify version).
- CAN bus (see ISOBUS, standard ISO 11783).
- Serial communication: USB (specify version) / RS-232 / RS-422 / RS-485 / PCI Express / Pmod / FMC - VITA57 / SPI / I²C / MIL-STD-1553 / others.

3.6. Indicators of Maintainability

Possible maintainability indicators are:

- Time to Fix Defects.
- Source code under a version control system.
- Well-established programming language.
- Automated build environments.
- Availability of an issue tracking system.
- Availability of “debugging modes” and tools.
- Availability of static and dynamic code analysis tools.
- Definition of a source tree structure.
- Automated documentation system.
- Availability and observance of a coding style guide.
- Availability of required and optional software dependencies (type of license, pricing, maintenance / development status).

3.7. Indicators of Marketability

For every instantiation of a product or service based on a software-defined GNSS receiver, managers should identify the (minimal) viable product, for which the organization will be continuously delivering (minimal) marketable features (MF) to create or maintain a (minimal) marketable product or service. From those definitions, possible marketability indicators are:

- Business impact: Savings obtained from the product or service with respect to a traditional (*i.e.*, integrated circuit based) approach.
- Defect Ratio: Percentage of the total MF which are defects.
- Work In Progress (WIP): From a cumulative flow diagram (CFD), compute the ratio between the MF WIP slope and closed MF slope. Both slopes should be the same to ensure optimal WIP limits are in place. That means the ratio should be 1. Forcing developers’ personal WIP limit to 1 is ideal, beyond 3 is chaotic.
- Delivery Frequency: From a CFD, compute the closed-issues slope. The slope should be either constant or increasing, never decreasing. Deployments (or licenses sold) per period could provide a good measure of how often the organization delivers.
- Throughput: Number of MF completed.
- Demand versus throughput balance: Open issues divided by target issues. This ratio should be maintained or decreasing, never increasing.

- Variability: From a control chart, compute the standard deviation of cycle and lead times. The objective is to narrow them.
- Productivity effectiveness: Ratio between current hours spent in the value stream and total hours paid.
- Productivity efficiency: Ratio between expected hours per MF and actual spent hours per MF.
- Knowledge capture: Number of final technical reports, which include recommendations, problems encountered, failure, success, learned lessons, etc. These documents have no value unless further accessed and re-used.

3.8. Indicators of Portability

Possible portability indicators are:

- Supported processor architectures:
 - i386: Intel x86 instruction set (32-bit microprocessors).
 - x86_64/amd64: the 64-bit version of the x86 instruction set, originally created by AMD and implemented by AMD, Intel, VIA and others.
 - armhf: ARM hard float, ARMv7 + VFP3-D16 floating-point hardware extension + Thumb-2 instruction set and above.
 - arm64: ARM 64 bits or ARMv8.
 - Support of GPU offloading (define vendor/model).
 - Support of FPGA offloading (define vendor/model).
 - Other (define).
- Supported operating systems:
 - GNU/Linux: specify distributions and versions.
 - Mac OS X: specify versions.
 - Microsoft Windows: specify versions.
 - Real Time Operating System (specify).
 - Others (define) / None (bare metal program).
- Other software dependencies (define).
- Minimal memory and storage requirements.

3.9. Indicators of Popularity

Possible popularity indicators are:

- Number of users / customers / licenses sold.
- Traffic to the site, measured with a counter installed on site and traffic analysis systems, such as Google Analytics.
- Size of users community.
- Number of references / citations in scientific papers.

3.10. Indicators of Reliability

Reliability is a concept that encompasses service continuity (and thus related to satellite availability and indicators of Sec-

tion 3.2), accuracy (Section 3.1), and integrity. The latter requires the definition, for each measurement of interest, of an *alert limit* (defined as the error tolerance not to be exceeded without issuing an alert), a *time to alert* (the maximum allowable time elapsed from the onset of the navigation system being out of tolerance until the equipment enunciates the alert), the corresponding *integrity risk* (probability that, at any moment, the position error exceeds the alert limit) and *protection level* (statistical bound error computed so as to guarantee that the probability of the absolute position error exceeding said number is smaller than or equal to the target integrity risk). Possible reliability indicators are:

- Percentage of false and missed alerts.
- Availability of receiver autonomous integrity monitoring (RAIM) mechanisms:
 - Fault detection (requires ≥ 5 in-view satellites).
 - Fault detection and exclusion (requires ≥ 6 in-view satellites).
 - RAIM prediction tools.
- Availability of mechanisms providing robustness against RF interferences and multipath:
 - Out-of-band rejection of RF interferences.
 - In-band rejection techniques for continuous wave, pulsed, and wideband interferences.
 - Countermeasures against spoofing, meaconing, and fake assisted and differential data.
 - Spatial diversity: Fixed / Controlled Reception Pattern Antennas.
- Deployment of network security and data integrity mechanisms.
- Availability of GNSS signal authentication mechanisms.
 - Probability of failure.
 - Time to authentication.
- Safety-critical software certifications (e.g., DO-178B).

3.11. Indicators of Repeatability

Possible repeatability indicators are:

- Stand-alone receiver's static positioning precision. Metrics are the same than in Table 1, where the standard deviations are now computed as:

$$\sigma_E^{(precision)} = \sqrt{\frac{1}{L-1} \sum_{l=1}^L (E[l] - \bar{E})^2} \quad (2)$$

where $\bar{E} = \frac{1}{L} \sum_{l=1}^L E[l]$ is the mean of all the E coordinates of the obtained positioning solutions, $E[l]$ are the East coordinates of the obtained positioning solutions, and L is the number of available position fixes. Similar expressions can be defined for the North and Up coordinates.

- Stand-alone receiver's dynamic positioning precision. Same metrics than in Table 1 and using definitions as in (2), where now measurements and the reference will have a time index.
- Differential GNSS static and dynamic positioning precisions.
- Average convergence times to sub-metric precision.

3.12. Indicators of Reproducibility

Possible reproducibility indicators are:

- Meet the requirements of *reproducible builds*¹, a set of software development practices which create a verifiable path from human readable source code to the binary code used by computers. This includes [16]:
 - The build system needs to be made entirely deterministic: transforming a given source must always create the same result.
 - The set of tools used to perform the build and more generally the build environment should either be recorded or pre-defined.
 - Users should be given a way to recreate a close enough build environment, perform the build process, and verify that the output matches the original build.

Several tools are now available to ensure reproducible builds. Some examples are provided in the referred website.

- Availability of unique identifiers for each source code snapshot.
- Availability of a Digital Object Identifier (DOI) for source code releases.
- Uniquely identifiable and reportable receiver configuration.

3.13. Indicators of Scalability

Possible scalability indicators are:

- Quasi-linear acceleration with the number of processors available in the computing platform.
- Arbitrarily scalable receiver's software architecture: unlimited addition of new GNSS signals and algorithms.
- Arbitrarily scalable configuration system.
- Maximum number of concurrent users.

3.14. Indicators of Testability

Possible testability indicators are:

- Availability of a testing framework, with the following desirable features (see [17] for details):

- Tests should be independent and repeatable.
- Tests should be well organized and reflect the structure of the tested code.
- Tests should be portable and reusable.

- Definition of a logging system allowing:
 - Setting up of severity levels and verbose modes for messages.
 - Setting up of conditional / occasional logging.
- Flexible configuration mechanism, allowing to set the receiver in the states described in Section 3.2.
- Definition of a profiling system.

3.15. Indicators of Openness

Possible openness indicators are:

- Software released under a free and open source license.
 - Allowing derivative works under the same license terms.
 - Allowing its commercial usage.
 - Dual licensing schemes.
- Availability of a technical report on algorithms and parameters used for:
 - Signal conditioning (possible digital down-conversion, filtering, decimation, sample format).
 - Signal acquisition.
 - Signal tracking.
 - Demodulation/decoding of navigation message.
 - PVT computation.
- In case of assisted / differential GNSS, reporting of the accessibility of the assistance / differential sources and nature of the delivered data.

3.16. Indicators of Usability

Possible usability indicators are:

- Availability of a (versioned) User Manual.
- Availability of a (versioned and documented) application user interface (API).
- Availability of graphical user interface.
- Availability of accessibility mechanisms for users who experience disabilities.
- Availability of mechanisms for remote operation.
- Availability of interfaces with other programming languages.
- Availability of user documentation: tutorials, detailed *howtos*, user stories, etc.

For the project, product or service in which the software receiver has a role:

- Website of project, product or service.

¹See <https://reproducible-builds.org>

- Availability of professional help desk support services.
- Availability of communication channels with other users and the development team.
 - Public mailing list.
 - Presence in public IRC channels.
 - Presence in social networks.

For the computing platform executing the software receiver:

- For physical devices:
 - Computer form factor, shape, size and weight.
 - Power consumption / battery autonomy.
 - Degrees of protection from solid objects and liquids (IP65 / IP67).
 - Temperature / humidity / vibration operative ranges.
- For cloud-based services:
 - Input/output data throughput requirements.
 - Computational and memory bandwidth requirements.
 - Connection to third parties in case of assisted or differential GNSS.

4. CONCLUSIONS

This paper is an attempt to answer the question: “How good is a particular embodiment of a software-defined GNSS receiver?” In order to ensure a comprehensive approach, the authors proposed a disparate list of design forces, or dimensions in which a given GNSS software-defined receiver implementation can improve – namely: accuracy, availability, efficiency, flexibility, interoperability, maintainability, marketability, portability, popularity, reliability, repeatability, reproducibility, scalability, testability, openness and usability. Their definitions were then used to generate an assortment of indicators (procedures, metrics and check points) that can be used as a basis when defining key performance indicators in contexts involving a software-defined GNSS receiver. The discussion continues at <http://gnss-sdr.org/design-forces/>

5. REFERENCES

- [1] J. B. S. Teasley, “Summary of the initial GPS Test Standards Document: ION STD-101,” in *Proc. 8th International Technical Meeting of the Satellite Division of The Institute of Navigation*, Palm Springs, CA, Sep. 1995, pp. 1645–1653.
- [2] Institute of Navigation, *ION STD 101 recommended test procedures for GPS receivers. Revision C*, Manassas, VA, 1997.
- [3] Agilent Technologies, *GPS Receiver Testing Application Note*, Santa Clara, CA, 2009.
- [4] National Instruments, *GPS Receiver Testing*, Austin, TX, 2016.
- [5] Rohde & Schwarz, *GPS, Glonass, Galileo Receiver Testing Using a GNSS Signal Simulator*, Munich, Germany, 2012.
- [6] Spirent, *How to construct a GPS / GNSS Test Plan*, Crawley, UK, 2016.
- [7] M. Avital, “The generative bedrock of open design,” in *Open design now: why design cannot remain exclusive*, B. van Abel, L. Evers, R. Klaassen, and P. Troxler, Eds., pp. 48–58. BIS Publishers, Amsterdam, The Netherlands, 2011.
- [8] G. Doran, “There’s a S.M.A.R.T. way to write management’s goals and objectives,” *Management Review*, vol. 70, no. 11, pp. 35–36, 1981.
- [9] G. Petit and B. Luzum, Eds., *IERS Conventions (2010)*, Verlag des Bundesamts für Kartographie und Geodäsie, Frankfurt, Germany, 2010, IERS Technical Note 36. ISBN 3-89888-989-6.
- [10] NovAtel Inc., *APN-029 GPS Position Accuracy Measures*, Calgary, Canada, 2003.
- [11] C. Hay, “Standardized GPS simulation scenarios for SPS receiver testing,” in *Proc. IEEE/ION Position, Location And Navigation Symposium*, San Diego, CA, Apr. 2006, pp. 1080–1085.
- [12] International GNSS Service (IGS), RINEX Working Group and Radio Technical Commission for Maritime Services Special Committee 104 (RTCM-SC104), “RINEX - The Receiver Independent Exchange Format. Version 3.03,” Tech. Rep., July 2015, Available online: <ftp://igs.org/pub/data/format/rinex303.pdf>. Accessed: January 30, 2017.
- [13] ION GNSS SDR Standard Working Group, “Global Navigation Satellite Systems Software Defined Radio Sampled Data Metadata Standard. Revision 0.1 (Initial Draft),” Tech. Rep., Jan. 2015, Available online: <https://github.com/IonMetadataWorkingGroup>.
- [14] The Radio Technical Commission for Maritime Services, *Differential GNSS (Global Navigation Satellite Systems) Services - Version 3*, Arlington, VA, Feb. 2013.
- [15] The Radio Technical Commission for Maritime Services, *RTCM 10410.1 Standard for Networked Transport of RTCM via Internet Protocol (Ntrip), Version 2.0 with Amendment 1*, Arlington, VA, Jun. 2011.
- [16] J. Bobbio, “How to make your software build reproducibly,” in *Chaos Communication Camp*, Mildenberg, Germany, 2015.
- [17] J. Whittaker, J. Arbon, and J. Carollo, *How Google tests software*, Addison-Wesley, Westford, MA, 2012.