INSPIRe

Functional and software design and Test Specifications – GPS M(G)RAIM

Prepared for:

Prepared by:	GMV NSL INSPIRe TEAM
Approved by:	M Pattinson (GMV NSL)
Authorized by:	T Richardson (GMV NSL)

Code:	INSPIRe-GMVNSL-Spec2.1-v1.0
Version:	1.0
Date:	December – 2022

Change Record

Issue / Rev	Date	Change Record	Authors
v1.0	19/12/2022	The first version of the deliverable submitted ESA	GMV INSPIRe Team

Table of Contents

List of Tables and Figures

1 INTRODUCTION

1.1 Purpose

This document is deliverable Spec2.1 of the INSPIRe project, titled 'Functional and Software design and test specification GPS M(G)RAIM. This deliverable document is an output for WP2 providing which includes the

- Functional specification and associated test specification for single frequency GPS MRAIM and MGRAIM
- Software design specification for implementing the GPS MRAIM and MGRAIM algorithms
- Software test specifications for each module for integration and at the system level

1.2 Scope

Following the background information presented in Section 1, the layout of the remainder of the document is as follows:

- Section 2 contains a list of applicable and reference documents
- Section 3 provides an overview of the algorithm design
- Section 4 provides an overview of the functional and software requirements
- Section 5 provides a description of the test plan to be executed to validate the systems implementing the Architecture and the resulting outcomes

1.3 Definitions and Acronyms

1.3.1 Definitions

Concepts and terms used in this document and need defining are included in the following table:

Table 1-1 Definitions

Concept / Term	Definition
MG-RAIM	Maritime General-RAIM: integrity algorithm designed for the configuration of single constellation, single frequency and without augmentation
MRAIM	MRAIM: integrity algorithm designed for a configuration of potential Multi-constellation, multiple frequencies, and with/without augmentation

1.3.2 Acronyms

Acronyms used in this document and need defining are included in the following table:

Table 1-2 Acronyms

Acronym	Definition
AL	Alert Limits
ARAIM	Advanced Receiver Autonomous Integrity Monitoring
CDF	Cumulative distribution function
DFMC	Dual Frequency Multiconstellation
DGNSS	Differential GNSS
DGPS	Differential GPS
DOP	Dilution of Precision

ECAC	European Civil Aviation Conference
EGNOS	European Geostationary Navigation Overlay Service
ESA	European Space Agency
FD	Fault Detection
FDE	Fault Detection and Exclusion
GBAS	Ground-Based Augmentation System
GEAS	GNSS Evolutionary Architecture Study
GLONASS	GLObal NAvigation Satellite System
GNSS	Global Navigation Satellite System
GPS	Global Positioning System
GRAD	GLA Research and Development
GSA	European GNSS Agency
HAL	Horizontal alarm Limit
HDOP	Horizontal Dilution of Precision
HMI	Hazardous Misleading Information
HPE	Horizontal Position Error
HPL	Horizontal Protection Level
IALA	International Association of Marine Aids to Navigation and Lighthouse Authorities
ICAO	International Civil Aviation Organisation
IEC	International Electrotechnical Commission
INSPIRe	Integrated Navigation System-of-Systems PNT Integrity for Resilience
IMO	International Maritime Organisation
IR	Integrity Risk
ISM	Integrity Support Message
LPV	Localizer Performance with Vertical guidance
MHSS	Multiple Hypothesis Solution Separation
MOPS	Minimum Operational Performance Standards
MGRAIM	Maritime General RAIM
MRAIM	Maritime RAIM
MSC	Maritime Safety Committee
MSR	Multi-system shipborne receiver
N/A	Not Applicable
NLOS	Non-Line of sight
NPA	Non-Precision Approach
PFA	Probability of False Alarm
PL	Protection Level
PHMI	Probability of Hazardously Misleading Information
PMD	Probability of Miss detection
PNT	Positioning Navigation and Timing
PVT	Position, Velocity and Time
RAIM	Receiver Autonomous Integrity Monitoring
RTCA	Radio Technical Commission for Aeronautics
RTK	Real-time kinematic positioning
SARPS	Standards and Recommended Practices
SBAS	Satellite Based Augmentation System
SIS	Signal in Space
SOLAS	Safety at Life at Sea
TBC	To Be Confirmed
TTA	Time to Arrival
VAL	Vertical alarm Limit

VHF	Very High Frequency

2 REFERENCE

2.1 Applicable Documents

The following documents, of the exact issue shown, form part of this document to the extent specified herein. Applicable documents are those referenced in the Contract or approved by the Approval Authority. They are referenced in this document in the form [AD.x]:

Table 2-1 Applicable Documents

Ref.	Title	Code	Version	Date
1.	INSPIRe Technical Proposal, Taylor Airey	T-062-001-02 Part 1	-	June 2022
2.	INSPIRe Management Proposal, Taylor Airey	T-062-001-02 Part 2	-	June 2022
3.	INSPIRe Proposal GMV	GMV 10842/21	V2/21	

2.2 Reference Documents

Although not part of this document, the following documents amplify or clarify its contents. Reference documents are those not applicable and referenced within this document. They are referenced in this document in the form [RD.x]:

Table 2-2 Reference Documents

Ref.	Title	Code	Version	Date
[RD.1]	INSPIRe Alg2.1 - Algorithm documentation (GPS M(G)RAIM)	INSPIRe-GMVNSL-Alg2.1-v1.0	1.0	10/2022
[RD.2]	IEC, 'Maritime navigation equipment - GNSS, part 1: GPS'	*IEC 61108-1*	-	2003
[RD.3]	EGNOS Message Server (EMS) User Interface Document	E-RD-SYS-E31-011-ESA	2.0	05/11/2004

3 ALGORITHM DESIGN OVERVIEW

The algorithm design functional and software architecture is described in [RD.1]. A high-level description is presented here for completeness.

The maritime integrity algorithm proposed for WP2 is based on a Classical RAIM algorithm, used for decades in maritime receivers, that is applied to the overall solution and to measurement subsets to ensure the fault detection (and exclusion if needed) capabilities.

The algorithm is based on two sequential steps once the position is calculated with all the available satellites.

- Availability check: A set of simple checks are applied to determine whether the derived all-in-view solution is suitable for navigation as defined by the maritime receiver specifications [RD.2]. If any of these tests fail, then an integrity alarm is raised.
- Fault detection: If all the previous test meets the defined conditions, a Chi-Squared test is performed to identify a fault in the positioning estimation.
- Geometry screening: For each potentially faulty element of the navigation solution, a subset navigation solution is formed by eliminating the faulty element from the all-in-view solution. Then, availability checks are performed to check if the remaining solution would pass, to determine if the detection capability of the solution is sufficient.

Figure 3-2 provides a high-level view of the rationale, inputs outputs and relationships of each of the functions detailed.

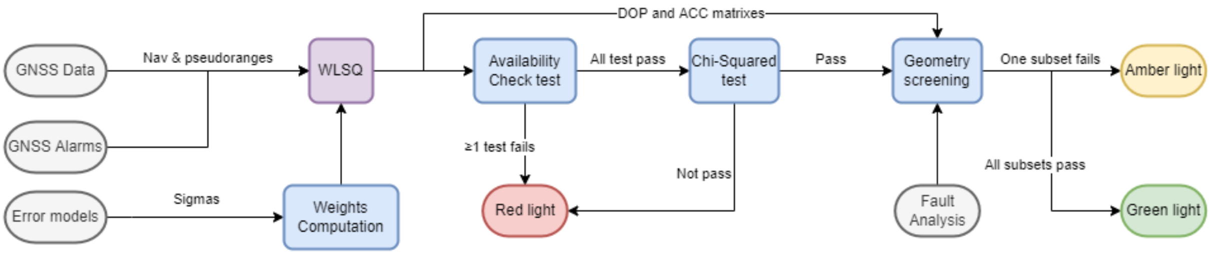

Figure 3-1 M-RAIM conceptual flowchart

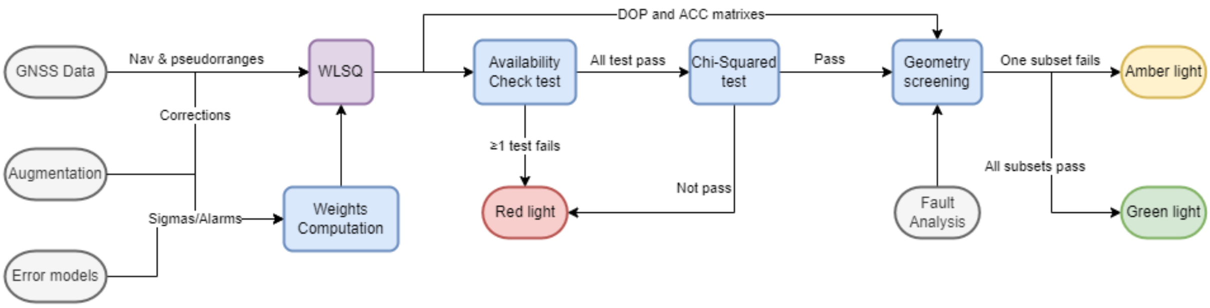

Figure 3-2. MG-RAIM conceptual flowchart

Where:

- Grey bubbles are the required inputs.
- Purple box is the PVT engine which computes the user estimated state vector.
- Blue boxes are the functions involved in the integrity algorithm
  - Weights computation: computes satellite error models considered as nominal for the maritime environment.
  - Availability Check Test: a function that checks if the current solution is suitable for maritime in terms of expected accuracy.
  - Fault Detection, Chi-Squared test: a function that detects if the solution complies with the nominal error models.
  - Geometry screening: a function that identifies the potential failure and performs additional tests to measurements subsets.

- Red, amber and green bubbles are output information provided to mariners

NB Fault Detection and Exclusion (FDE) may be used by some future refinement of this process and may be a user-set option, therefore the scheme will include an iteration loop after an amber light is raised, excluding the identified fault from the solution and starting the WLSQ module again without that measurement.

4 SPECIFICATION OF REQUIREMENTS

This section presents functional, non-functional, and safety requirements for the main elements of the PVT Algorithm architecture.

4.1 Requirement nomenclature

For each requirement the following characteristics are specified:

- ID — A unique identifier: it is composed of a string and a numeric part.
- Description — The content of the requirement.
- Trace-back — Points to source material from which the requirement has been derived.

Note: All specified requirements are to be considered mandatory unless otherwise stated.

The unique requirement ID complies with the following format:

[Category].[Group].[###]

Where:

- [Category] can be
  - FUN - functional requirements.
  - INT - interface requirements
- [Group] can be
  - ALG - requirements related to the Algorithm
  - SW - requirements related to the Software

- [###] is a numeric incremental identifier for requirements which have the same [Category]. [Group] prefix.

The requirement characteristics are presented in paragraph format, as demonstrated in Table 4-1.

ID	[Category].[Group].[###]
Description	A concise description of the requirement

Table 4-1: Requirement specification format

4.2 Algorithm Functional Specification

FUN. ALG.001

The algorithm shall process the GNSS input data, as specified in the table below, prior to position and integrity calculations.

Input data	Guidance
GNSS constellation ID	GPS
GNSS frequency ID	L1

Code Observation	Measurement
Carrier phase Observation	Measurement
Navigation Data	Decoded messages

FUN.ALG.002

The algorithm shall Generate a snapshot weighted least squares 3D position solution using code measurements

Req ID	Guidance
FUN.ALG. 002.01	Be able to do a standalone solution for GPS L1 only
FUN.ALG. 002.02	Be able to do an SBAS corrected GPS L1 solution

FUN.ALG.003

The algorithm shall reject satellites that are flagged as ‘do not use’, while in SBAS Mode.

FUN.ALG.004

The algorithm shall use UDREI values from SBAS messages to weight orbit and clock and Ionosphere contribution to error budgets, while in SBAS Mode.

FUN.ALG.005

The algorithm shall use GIVEI values from SBAS messages to Ionosphere contribution to error budgets, while in SBAS Mode.

FUN.ALG.006

The algorithm shall use the navigation message broadcast URA value to weight orbit and clock contribution to error budgets, while in GPS Mode.

FUN.ALG.007

The algorithm shall use Configurable values for other contributions to the error budget.

FUN.ALG.008

The algorithm shall use set weights using error budget values.

FUN.ALG.009

The algorithm shall perform an availability check of the solution to check HDOP, GDOP, 95% horizontal accuracy and the number of satellites against thresholds

FUN. ALG.010

The algorithm shall raise a red-light integrity indicator for that epoch if the availability check fails

FUN. ALG.011

The algorithm shall perform fault detection for the solution using measurement residuals and check against a threshold

FUN. ALG.012

The algorithm shall perform geometry screening on selected subsets of satellites to check HDOP, GDOP, 95% horizontal accuracy and number of satellites against thresholds

FUN. ALG.013

The TOOL shall raise traffic light indicators based on the following:

	Traffic light Flag Indicator	Guidance
FUN.ALG. 013.01	RED	An indicator flag is raised for that epoch if a fault is detected
FUN.ALG. 013.02	AMBER	An indicator flag is raised if geometry screening fails for any subset
FUN.ALG. 013.03	GREEN	An indicator flag is raised if availability check, fault detection and geometry screening do not raise any issues.

FUN. ALG.014

The algorithm shall set the output integrity flag to red if the red-light integrity indicator was raised at any point in the previous applicable period (configurable).

FUN. ALG.015

The algorithm shall set the current output integrity flag to the integrity indicator for the current epoch if the red-light integrity indicator was not raised at any point in the previous applicable period (configurable).

INT. ALG.016

The algorithm shall provide the following output data:

Req. ID	Output data	Guidance
INT.TOOL. 016.01	3D position	WGS84 coordinates
INT.TOOL. 016.03	GNSS timestamp	
INT.TOOL.1016.05	Geometry Screening Indicator Flags	

4.3 Software Functional Specification

FUN. SW.001

The software shall Implement an algorithm with the functional requirements listed above.

FUN. SW.002

The software shall run in post-processing and not a Real Time data stream

FUN. SW.003

The software shall read the configuration settings from the configuration file.

FUN. SW.004

The software shall read and save the RINEX Observation file.

FUN. SW.005

The software shall read and save the RINEX Navigation file.

FUN. SW.006

The TOOL shall read and save the SBAS message.

FUN. SW.007

The TOOL shall apply carrier smoothing of the code for single-frequency measurements (the smoothing constant shall be configurable).

FUN. SW.008

The software shall read the smoothing constant from the configuration parameters

FUN. SW.009

The software shall read the elevation mask from the configuration parameters, removing observation for the satellite below the mask.

FUN. SW.010

The software shall apply RTCA tropospheric correction model.

FUN. SW.011

The software shall apply the Klobuchar model for ionospheric correction, in GPS only mode

FUN. SW.012

The software shall make the SBAS mode configurable to be enabled or disabled

FUN. SW.013

The software when operated in SBAS mode shall:

Req ID	Guidance
FUN. SW.013.01	Remove satellites that have 'do not use' condition in UDRE / GIVE
FUN. SW.013.02	Apply fast/slow clock corrections
FUN. SW.013.03	Apply ionospheric GIVD corrections

FUN. SW.014

The software shall make the following configurable

Req ID	Guidance
FUN. SW.014.01	Thresholds for availability check configurable
FUN. SW.014.02	False alarm rate for fault detection configurable
FUN. SW.014.03	Thresholds for geometry screening configurable

FUN. SW.015

The software shall configurable a time period for keep red flag

FUN.SW.016

The software shall execute the position computation in both GPS mode and SBAS mode

INT.SW.017

The software shall process one epoch of observations and save the results. The Position Results saved should include but not limited to the following:

Position Results Output
▪ Time ▪ Estimated Position (XYZ) ▪ Estimated position (LLH) ▪ Integrity flag (Green, Amber, Red) ▪ HDOP ▪ VDOP ▪ GDOP ▪ Number of Satellites used ▪ Computed test statistics ▪ FD threshold

5 TEST PLAN

This section contains the test cases that have been created to test the operation of INSPIRe algorithm architecture described in the deliverable Alg2.1. Within describes which tests are to be executed and provide a high-level description of the purpose of the test along with required data and test tools.

All test cases shall be independent; no test case will rely on the output of another test case. This has the benefit that the tests can be executed in any order, except for the Installation test procedure which will be run first, and that the failure of a given test will not prevent the running of other tests.

5.1 Test Data

The data and configuration settings at all processing stages are stored in files. Sample data sets and configurations for testing will be created during the development of the algorithm. These files will be archived with documented filenames.

Test data consists of:

- GNSS RINEX version 3.04 format
  - Observation File
  - Navigation File
- • EGNOS data

  Table 5-1 shows the EGNOS messages dataset format.

Table 5-1 -Archived EGNOS message data sets

Data Set	Format	Rate	Periodicity of Publication
EGNOS messages	EGNOS Message Server (EMS)	1 Hz	1 hour

The EMS format has been defined by ESA for the provision of EGNOS messages. EMS format 2.0 is described in [RD.3]. The existing GMV NSL SBAS Message (e.g. EGNOS) decoder uses the ESA-defined EMS format, and produces the decoded messages in *.csv files. The EMS server allows access to the EGNOS messages offline, in the form of an ASCII file.

5.2 Test Environment

The test will be carried out standalone from the GMVNSL network and will use a PC or laptop as the “diagnostics” machine running Windows OS

5.3 List of Test Cases

Each test case will be defined based on a Case ID with next format:

TC - Prefix to indicate this is a Test Case Identifier.

[Function] - is the name of the software component under testing

[ID#] – sequential numbering for each case.

Table 5-2 below identifies a set of test cases proposed. To demonstrate that the algorithm is compliant with its system, it is necessary to provide a trace of the system requirements in the test activities. Each requirement will be traced to the test that verifies it. In some instances, requirements may need more than one test to demonstrate compliance (e.g., if the requirement has two or more ‘parts’ to it). In such cases, a comment will be included in the trace to identify the aspects of the requirement that each test is responsible for demonstrating.

Not all functional specifications will be traceable to a functional test execution; these may require inspection of hardware and/or technical documentation.

Table 5-2 Test Cases

Test CaseID	Objective/Test sets	Description	Success Criteria	Traceability Requirements
TC.SW.01	To check that files are read, and a solution is generated as expected	Execution of the positioning software using a nominal configuration which includes: • Archived real data (GPS L1 - RINEX Observation(*.obs) and Navigation (*.nav)), • Faults: = NONE, • Smoothing constant: 100s	• Population of the output parameter • Manually check the consistency of loaded data • a valid solution, green flag, horizontal position errors < 10m • Solution output can be compared with commercial/open sources PVT processing software to check that the position is as expected, (e.g., solution, number of satellites, DOP values, etc)	FUN. ALG.001 FUN.ALG.002 FUN.ALG. 020.01 FUN.ALG. 013.03 INT. ALG.016 FUN. SW.001 FUN. SW.002 FUN. SW.003 FUN. SW.004 FUN. SW.005 FUN. SW.010 FUN. SW.011 FUN.SW.016 FUN.SW.017
TC.SW.02	To demonstrate that the availability check is working and HDOP threshold is configurable.	Same as TC.SW.01 with the HDOP threshold configuration set to very small value	The output integrity flag is red	FUN.ALG.009 FUN. ALG.010 FUN. SW.014.01
TC.SW.03	To demonstrate that the availability check is working and GDOP threshold is configurable	Same as TC.SW.01 with the GDOP threshold configuration set to very small value	The output integrity flag is red	FUN.ALG.009 FUN. ALG.010 FUN. SW.014.01
TC.SW.04	To demonstrate that the availability check is working, and the Horizontal accuracy threshold is configurable	Same as TC.SW.01 with the Horizontal accuracy threshold configuration set to a very small value	The output integrity flag is red	FUN.ALG.009 FUN. ALG.010 FUN. SW.014.01
TC.SW.05	To demonstrate that the availability check is working, and elevation mask is configurable	Same as TC.SW.01, but increase configured elevation mask so that have a minimum of 4 satellites in view	• The number of satellites used is lower because low-elevation satellites are removed • The output integrity flag is red when 4 satellites are in view	FUN.ALG.009 FUN. ALG.010 FUN. SW.009
TC.SW.06	To demonstrate the geometry screening check is working and HDOP threshold is configurable	Same as TC.SW.01 but set HDOP threshold to only just larger than the maximum all-in-view HDOP from any epoch in the test data	The output on the epoch that has the highest all-in-view, HDOP should see an amber flag	FUN. ALG.012 FUN.ALG. 013.02 FUN. SW.014.03
TC.SW.07	To demonstrate the geometry screening check is working and GDOP threshold is configurable	Same as TC.SW.01 but set GDOP threshold to only just larger than the maximum all-in-view GDOP from any epoch in the test data	The output on the epoch that has the highest all-in-view GDOP should see an amber flag	FUN. ALG.012 FUN.ALG. 013.02 FUN. SW.014.03

TC.SW.08	To demonstrate the geometry screening check is working and horizontal accuracy threshold is configurable	Same as TC.SW.01 but set the horizontal accuracy threshold to only just larger than the maximum all-in-view horizontal accuracy from any epoch in the test data	The output on the epoch that has the highest all-in-view Horizontal accuracy should see an amber flag	FUN. ALG.012 FUN.ALG. 013.02
TC.SW.09	To check that the fault detection function can detect the fault and that the red integrity flag is raised	Same as TC.SW.01 but inject a large bias fault (100m) on one satellite for one second, and again 10 seconds later. Assuming that the time configuration for the red flag is 6 seconds	• Position errors same as in TC.SW.01 except on the two affected epochs when they will be much larger. • The integrity flag is green on all epochs except on the epochs where the fault is added and for 6 seconds after each fault when it is red	FUN. ALG.011 FUN.ALG. 013.01 FUN. ALG.014 FUN. ALG.015
TC.SW.10	To check that the fault detection function can detect the fault and that the red integrity flag is raised and kept for a certain time	Same as TC.SW.09 except for config for the red flag time period set to 12 seconds	The integrity flag should be green except for the period when the first fault injected has started and lasting until 12 seconds after the second fault is injected when it is red	FUN. ALG.011 FUN.ALG. 013.01 FUN. ALG.014 FUN. ALG.015 FUN. SW.015
TC.SW.11	To check that the URA is being considered in weighting	Same as TC.SW.01 but change URA values for all navigation messages to be bigger	• The position solution should be identical to test TC.SW.01 • The test statistic on each epoch should be smaller and the horizontal accuracy value bigger	FUN.ALG.006 FUN.ALG.008
TC.SW.12	To check that the error budgets are configurable and are used	Same as TC.SW.01 but increase error budget values at all elevations	• The position solution should be identical to test TC.SW.01 • The test statistic on each epoch should be smaller and the horizontal accuracy value bigger	FUN.ALG.007 FUN.ALG.008
TC.SW.13	To demonstrate that the Software can produce a SBAS solution	Same as TC.SW.01 but now generating an SBAS solution	• The output should produce a valid solution, green flag errors < 10m • The test statistic and accuracy value should be different to the GPS case (even for the same satellites) • Solution output can be compared with commercial/open sources PVT processing software to check that the position is as expected • Manually check the consistency of loaded EGNOS data	FUN.ALG. 020.02 FUN.ALG.004 FUN.ALG.005 FUN.ALG.008 FUN. SW.006 FUN. SW.012 FUN. SW.013.02 FUN. SW.013.03 FUN.SW.016
TC.SW.14	To demonstrates that satellites marked "do no use" by EGNOS are removed	Same as TC.SW.13 but modify one of the SBAS messages for one satellite identified as 'do not use'	When the satellite is set 'do not use': • Manually check the consistency of Satellite information i.e. – the number of satellites is reduced by 1	FUN. SW.013.01

			• The test statistic and accuracy value should be different to the	
TC.SW.15	To demonstrate that these parameters are configurable	Same as TC.SW.13 but change 1. The smoothing constant to a lower value 2. The false alarm rate to a smaller value (e.g., $1e^{-6}$)	Compared to test 13, the output should produce: • Position estimates that are different • Test statistics thresholds that are larger	FUN. SW.007 FUN. SW.008 FUN. SW.014.02

End of Document