Available online at www.sciencedirect.com

ScienceDirect

Advances in Space Research xxx (xxxx) xxx

ADVANCES IN SPACE RESEARCH (a COSPAR publication)

www.elsevier.com/locate/asr

FocusPOD, a new operational geodesy tool

Jaime Fernández [a,*], Carlos Fernández [a], Javier Berzosa [a], Miguel Ángel Muñoz [a], Luning Bao [a], Marc Fernández [a], Oleksandr Ivanchuk [a], Sonia Lara [a], Eva Terradillos [a], Heike Peter [b], Muriel Pinheiro [c], Carolina Nogueira [d]

[a] *GMV AD., Tres Cantos, Spain*
[b] *PosiTim UG, Seeheim-Jugenheim, Germany*
[c] *ESA/ESRIN, Largo Galileo Galilei 1, I-00044 Frascati, Italy*
[d] *EUMETSAT, Eumetsat Allee 1, 64295 Darmstadt, Germany*

Received 23 November 2025; received in revised form 27 March 2026; accepted 28 March 2026

Abstract

FocusPOD is a modern astrodynamics and space-geodesy software library developed in C++ and Python by GMV to support operational and research-grade Precise Orbit Determination (POD) and geodetic applications. The system is built upon a modular, library-oriented architecture and a relational data model that strictly separates data structures from algorithms, enabling extensibility, reproducibility, and efficient large-scale processing. A comprehensive Python interface provides high-level orchestration and user-defined extensions while preserving the numerical performance of the C++ core and avoiding intermediate input/output operations.

FocusPOD has been operationally deployed within the Copernicus Precise Orbit Determination (CPOD) Service since 2023 for the generation of Sentinel orbit products. It delivers centimetre-level accuracy, with Near-Real-Time products routinely generated in less than five minutes and achieving ~3 cm 3D RMS for Sentinel-1/2 and ~1–2 cm radial RMS for altimetry missions. A dedicated reprocessing campaign of Sentinel-3A, Sentinel-3B, and Sentinel-6A demonstrates mean orbit differences of ~5 mm (3D RMS) and ~2–3 mm (radial RMS) with respect to combined solutions over multi-year intervals. Independent validation using Satellite Laser Ranging (SLR) confirms orbit accuracy at the 5–7 mm level in the radial component. These results demonstrate that *FocusPOD* provides a scalable, high-performance and operationally robust framework for GNSS-based LEO POD, while offering a unified foundation for future multi-technique geodetic processing and reference frame realization.

Keywords: Precise orbit determination; Copernicus; Sentinels; GNSS; SLR

1. Introduction

GMV has been actively involved in Precise Orbit Determination (POD) since the 1980s. Early contributions included GMV staff working at the European Space Operations Centre (ESOC), supporting the development of BAHN, a precise orbit determination system designed for the first European altimetry missions. By the mid-1990s, GMV became involved in the development and evolution of NAPEOS (Navigation Package for Earth Observation Satellites) (Springer et al., 2011), providing support to ESA across multiple enhancements throughout the 1990s and 2000s. Over the years, GMV has also contributed to several ESA and EUMETSAT missions requiring POD, including MetOp, SWARM, and more recently the Copernicus Sentinel missions.

In 2013, GMV was awarded a contract by ESA to develop and operate the Copernicus Precise Orbit Determination (CPOD) Service (Fernández et al., 2024),

* Corresponding author.
E-mail address: jfernandez@gmv.com (J. Fernández).

https://doi.org/10.1016/j.asr.2026.03.098
0273-1177/

Please cite this article as: J. Fernández, C. Fernández, J. Berzosa et al., *FocusPOD*, a new operational geodesy tool, Advances in Space Research, https://doi.org/10.1016/j.asr.2026.03.098

responsible for generating precise orbits and auxiliary products for Sentinel-1, Sentinel-2, Sentinel-3, and later Sentinel-6. Leveraging its extensive experience with NAPEOS, the CPOD Service was initially built upon NAPEOS technology and remained operational until the end of 2022.

In 2021, GMV initiated an internal Research and Development (R&D) activity aimed at developing a new POD software system. This decision was driven by the recognition that periodic technology renewal is essential. While continuous updates to physical and dynamical models are standard practice in scientific software, advances in information technology (IT) impose additional and increasingly stringent requirements. Although GMV devotes significant resources to R&D, it is fundamentally an engineering company focused on the development of complex operational software systems, rather than a purely academic research institution. Consequently, GMV's clients (e.g., ESA and EUMETSAT) increasingly demand the adoption of modern IT paradigms and software architectures.

Despite its well-established reputation in the POD community and its proven capability to achieve excellent orbit accuracy, NAPEOS faced growing limitations in meeting these evolving IT requirements. Challenges included the programming language ecosystem (with increasing demand for C++ and Python), integration within larger operational systems, flexible management of input/output and configuration files, interaction with databases, usability within Jupyter notebooks, high and even dynamic levels of configurability, and continuous pressure to improve computational performance in terms of memory usage and execution speed.

Many of these limitations stem from architectural design choices dating back to the mid-1990s, when multithreading, large memory availability, and tight integration with external software environments were not primary concerns. Over time, the accumulation of new operational requirements placed increasing strain on this legacy architecture.

In parallel, GMV develops not only POD systems but also software for flight dynamics, space traffic management, and mission analysis, collectively encompassing a broad range of space mechanics applications. In this context, it was also observed that institutions such as ESOC and the Jet Propulsion Laboratory (JPL) (Bertiger et al., 2020) have undertaken similar technology transitions, involving complete or near-complete re-implementations of legacy software using modern C++ and Python-based frameworks.

An additional factor influencing the decision to develop a new in-house software was related to licensing conditions. Owning the full software stack provides GMV with greater flexibility when developing and deploying operational systems for a wide range of international users.

The development of the new POD software was initiated with full awareness that POD is a mature scientific field, supported by several well-established software packages. Table 1 presents a selection of widely used POD tools within the POD and geodetic communities. While GMV's direct operational experience was largely centred on NAPEOS, these systems provided valuable inspiration in terms of scope and capabilities.

As a first step, GMV initiated the development of *FocusPOD*, targeting Low Earth Orbit (LEO) precise orbit determination using Global Navigation Satellite System (GNSS) observations, which is the focus of the present paper. At the same time, GMV's internal R&D roadmap is intentionally ambitious, aiming in the long term to achieve capabilities comparable to highly mature software packages of Table 1, particularly the capability of combining the four space-geodetic techniques.

The design of *FocusPOD* has been inspired by two specific software systems developed at ESA/ESOC. The first one is the General Orbit Determination and Optimization Toolkit (GODOT) (Mackenzie, 2021) which represents a significant step forward in flight-dynamics software through its highly generic, modular and extensible C++ library combined with a Python interface. GODOT inspired several technological aspects of *FocusPOD*, in particular its emphasis on library extensibility and the provision of a comprehensive Python API.

The second source of inspiration is ROBOD (Relational Orbit Determination) (Boomkamp, 2006), an internal development at the ESA/ESOC Navigation Support Office. ROBOD introduced a relational-database-style internal architecture, in which objects map to records and relationships are explicitly defined, ensuring uniqueness of information and avoiding redundant data storage. This design supports more efficient memory usage and computational performance. The data model of *FocusPOD* directly builds upon these concepts.

In summary, *FocusPOD* was developed to address the growing need for a modern, flexible, and maintainable astrodynamics and space-geodesy software library capable of supporting both operational-services and research-oriented activities. Its design reflects the evolution of software engineering practices over recent decades, with a particular emphasis on modularity, extensibility, and scalability.

Many established tools in orbit determination and geodesy were originally developed in FORTRAN 77 or FORTRAN 90 and, while scientifically robust, have progressively become more complex and difficult to maintain or adapt to new operational environments. *FocusPOD* is designed according to modern software engineering principles, leveraging C++17 for the computational core and Python 3 for high-level interaction and orchestration. Modern C++ provides access to a rich standard library and language features—including the Standard Template Library (STL), smart pointers, strong typing, inheritance, and generic programming—which reduce the amount of custom code required and facilitate safer memory

Table 1
Summary of selected POD and geodetic software systems.

Software	Institution	Primary scope	Measurement techniques	Estimation method	Language	Notable characteristics	Reference
Bernese (BSW)	AIUB/CODE	GNSS & LEO POD	GNSS, SLR	Batch WLS[a]	FORTRAN	GNSS network processing with solution in IGS. Support PPP processing.GNSS-based LEO POD in reduced dynamics and kinematic.	Dach et al. (2015)
NAPEOS/EPNS[b]	ESA/ESOC	GNSS & LEO POD	GNSS, SLR, DORIS	Batch WLS	FORTRAN	GNSS network processing with solution in IGS. GNSS + DORIS + SLR LEO POD in reduced dynamics.	Springer et al. (2011)
GHOST	DLR/GSOC	LEO POD	GNSS	Batch WLS, EKF[c]	C++	GNSS-based LEO POD in reduced dynamics with SLR validation. Include Batch WLS and EKF. Relative navigation.	Wermuth et al. (2010)
GipsyX/RTGx	JPL	GNSS & LEO POD	GNSS, SLR, DORIS, VLBI	EKF, SRIF[d]	C++, Python	Complete multi-technique space-geodesy SW.	Bertiger et al. (2020)
GROOPS	TU Graz	GNSS & LEO POD	GNSS, SLR	Batch WLS	C++	GNSS network and GNSS-based LEO POD processing using raw processing. Open source. It also supports gravity field recovery	Mayer-Gürr et al. (2021)
EPOS OC	GFZ	GNSS & LEO POD	GNSS, SLR, DORIS, VLBI	Batch WLS	FORTRAN	Complete multi-technique space-geodesy SW.	Neumayer et al. (2024)
GEODYN II	NASA/GSFC	POD & Geodesy	GNSS, SLR, DORIS, VLBI	Batch WLS	FORTRAN	Complete multi-technique space-geodesy SW.	NASA (2023)
ZOOM	CNES	LEO POD	GNSS, DORIS, SLR	Batch WLS	FORTRAN	GNSS + DORIS + SLR LEO POD in reduced dynamics.	Carrou (1986)
GINS/DYNAMO	CNES/GRGS	POD, PPP, Geodesy	GNSS, SLR, DORIS, VLBI	Batch WLS	FORTRAN	Complete multi-technique space-geodesy SW.	Marty et al. (2011)
GREAT	Wuhan University	GNSS & LEO POD	GNSS	Batch WLS, SRIF	C++	GNSS network processing.GNSS-based LEO POD.	Li et al. (2024)
PANDA	Wuhan University	GNSS & LEO POD	GNSS	Batch WLS	C/C++	Predecessor to GREAT.	Shi et al. (2008)
MADOCA	JAXA	GNSS	GNSS	Batch WLS, EKF	C/C++	GNSS network processing.	Kawate et al. (2023)
GSharp	GMV	GNSS	GNSS	Batch WLS, EKF	C++	GNSS network processing in real-time.	GMV (2023)

[a] Weight Least Square.
[b] ESA Precise Navigation Software, evolution of NAPEOS.
[c] Extended Kalman Filter.
[d] Square Root Information Filter.

management, clearer interfaces, and improved long-term maintainability. In addition, the C++ ecosystem offers mature third-party libraries for common functionalities including logging, configuration parsing, database access, and language bindings, as well as high-performance scientific computing.

Python complements the C++ core by offering a high-level environment that is widely adopted in scientific and engineering communities, with extensive support for numerical computation, data analysis, visualization, and emerging domains such as machine learning. Through dedicated C++–Python bindings, *FocusPOD* exposes core functionality directly to Python, enabling rapid prototyping of processing workflows, interactive analysis, and seamless integration with Python-based tools. This approach supports efficient development cycles and exploratory analysis, for example through Jupyter notebooks, while preserving the computational performance of the C++ core.

It is important to emphasize that the main contribution of *FocusPOD* does not lie in the introduction of new physical force models or observation models, which are well established in the field and continuously refined by the broader POD community. Instead, the innovation of *FocusPOD* resides primarily in its software architecture and internal design, which have been conceived to support long-term maintainability, extensibility, and operational robustness.

Although *FocusPOD* is architecturally conceived as a multi-technique geodetic platform, this article focuses exclusively on the GNSS-based LEO POD application, which is currently the most mature and operationally validated use case.

Currently, *FocusPOD* is used operationally by the CPOD Service (Fernández et al., 2024) to generate precise orbits (see Section 3). It is also employed within the Galileo Second Generation (G2G) System Test Bed (STB) to implement simulation capabilities for GNSS, SLR and Inter-Satellite Link (ISL) measurements. In addition, *FocusPOD* will be used by the Galileo Reference Centre (GRC) for the processing of VLBI observations and will be adopted by ESA´s LEO Position, Navigation and Timing (LEO-PNT) project to generate precise orbit and clock products for mission performance validation. Beyond these operational uses, *FocusPOD* has supported POD simulations in several ESA projects and is currently used to implement a simulator for advanced POD concepts applied to LEO mega-constellations.

The remainder of this paper is organized as follows. Section 2 describes the software architecture and design principles of *FocusPOD*. Section 3 details its operational application within the CPOD Service. Section 4 evaluates system performance through reprocessing experiments for the Sentinel 3A, Sentinel 3B, and Sentinel 6A missions, including an independent accuracy assessment using SLR observations. Finally, Section 5 summarizes the main findings and outlines perspectives for future developments.

2. Software design

FocusPOD is the commercial name for the POD and Geodesy applications of GMV MAORI (Multi-Purpose Advanced Orbit Reconstruction Infrastructure), the underlying C++/Python library described in this section. For practical purposes, *FocusPOD* and MAORI can be considered equivalent. However, MAORI also supports other domains, such as Flight Dynamics and Space Surveillance & Traffic Management (SSTM), which are beyond the scope of this paper. The name *FocusPOD* is used here to emphasize the POD and Geodesy aspects of the system.

FocusPOD was developed over an initial two-year period (approximately 15,000 h) by a team of 6–10 early-career engineers, following an agile development methodology based on Scrum.[1] Scrum was chosen to manage the complexity of the project, as the overall design of the library was not fully defined at the outset. Scrum is an iterative and incremental framework in which development is organized into short, time-boxed cycles (“sprints”), typically lasting 2–4 weeks, enabling frequent reassessment of requirements and design. Progress is driven by prioritized work items defined in a product backlog, and regular review and planning meetings facilitate continuous feedback, incremental delivery of functionality, and risk mitigation in complex software developments.

Only a few key design principles were established initially, and development was supported by the availability of NAPEOS, a FORTRAN-based software system, which enabled thorough validation of results.

The design principles guiding *FocusPOD* included: (1) developing a cohesive library rather than a collection of standalone programs; (2) clearly separating data structures from algorithms, as described by Raymond (2003); (3) ensuring that data structures could be extended for future applications by organizing data within a well-defined and maintainable data model; and (4) supporting a wide range of use cases. Furthermore, the new library was required to match the accuracy of NAPEOS while exceeding it in computational performance. With respect to accuracy, the physical models were preserved and carefully validated to ensure no degradation in quality. Performance improvements were instead achieved through architectural and algorithmic optimizations, notably by minimizing unnecessary input/output operations and by revising selected algorithms to improve their computational efficiency.

The Scrum methodology provided a structured yet flexible development process that supported incremental implementation, continuous testing, and regular reassessment of design choices. This iterative process allowed the team to explore alternative architectural and algorithmic solutions, evaluate their impact on accuracy and performance, and refine or redesign components when necessary. As a result, simplicity was preserved by revisiting design

[1] https://www.scrum.org/learning-series/what-is-scrum/.

decisions early, rather than introducing compensating workarounds at later stages.

2.1. High level architectural elements

2.1.1. Library oriented

One of the core objectives behind *FocusPOD* is to develop a multi-purpose software library rather than a collection of fixed, standalone programs. This approach ensures that all functionality is implemented as generically as possible, avoiding hard-coded assumptions tied to specific use cases such as POD. Instead, the library is designed to provide a high degree of user interaction and configurability, empowering users to tailor their own applications according to project-specific requirements.

This design philosophy is essential to supporting a wide range of projects, each with its own architectural and technological needs. Flexibility and generality are prioritized throughout the development process, enabling seamless integration into different systems. Applications can then be easily built using *FocusPOD* as a dependency, with well-defined interfaces that clearly handle both input configurations and output results.

2.1.2. Split data from algorithms

A key design principle in *FocusPOD* is the clear separation of data and algorithms. Library modules are designed to operate on well-defined, documented central data structures, with algorithms focusing solely on computation. This separation ensures that algorithms are not responsible for acquiring, organizing, or post-processing data in a hard-coded or inflexible manner.

This principle is essential to achieving the broader goal of flexibility and is deeply aligned with the UNIX philosophy as described by Raymond (2003). In particular, it embodies *the Rule of Representation* "fold knowledge into data, so program logic can be stupid and robust" and the *Rule of Separation* "separate policy from mechanism; separate interfaces from engines" which advocates decoupling the *what* from the *how*. In the context of *FocusPOD*, this means that algorithms remain agnostic to data sourcing and structuring, thereby promoting cleaner interfaces and broader applicability.

By decoupling data handling from algorithmic logic, the library allows users to control how data is sourced, prepared, and interpreted, enabling the same algorithms to be reused across diverse applications, including those not yet imagined. This level of abstraction not only enhances code maintainability but also makes the library more extensible, as future use cases can build on existing logic without requiring changes to core computations.

Furthermore, this separation encourages the development of more generic and composable code, where algorithms become reusable building blocks rather than tightly coupled components. It also facilitates testing, validation, and debugging by isolating concerns, another hallmark of robust software architecture.

2.1.3. Data model

A foundational design principle in *FocusPOD* is the organization of data within a clear, relational data model that reflects physical reality as faithfully as possible. The model includes, among others, entities such as satellites, ground stations, instruments, oscillators, and antennas. This approach naturally follows from the prior principle of separating data from algorithms. Since algorithms operate on structured data, the library must be responsible for ingesting, organizing, and circulating that data as required by each use case.

This consistency in representing physical reality is especially beneficial across diverse projects: while data formats or structures may vary by project, the underlying physical concepts remain the same. As a result, the data model of *FocusPOD* is designed to:

- Faithfully represent physical realities. This ensures long-term extensibility, allowing new data types to be integrated by maintaining their physical meaning and relationships. It also simplifies development by centralizing all data, avoiding fragmentation across the codebase.
- Provide intuitive and efficient access. Algorithms and users access data through well-defined interfaces, improving code reuse and modularity.
- Scale efficiently. The model is designed to support applications involving large volumes of data, such as geodetic processing (thousands of satellites, receivers and measurements), space debris cataloguing, and mega-constellation flight dynamics, without compromising performance.

Beyond conceptual fidelity, adopting a relational, entity-centric model yields the following practical benefits:

- Easy data reuse. Well-defined entities (such as satellites and instruments) act as shared components across pipelines, allowing algorithms and tools to work together without additional interface code.
- Portable persistence and interchange. Decoupling schema from storage allows the same data model to be serialized to multiple media (files, databases, or network Application Programming Interfaces – APIs) via pluggable I/O adapters, without touching algorithmic code.
- Scenario snapshotting and recovery. The entire *Scenario* (the class encapsulating the data model) can potentially be serialized to and reconstructed from a file, enabling checkpoint/restart, rollback, and fully reproducible reprocessing.
- Reproducibility and auditability. Versioned schemas and embedded metadata ensure deterministic and reviewable experiments, while complete provenance tracking preserves the inputs and configurations associated with each result.
- Testability and Continuous Integration (CI). Algorithms can be unit-tested against minimal in-memory tables, and I/O operations can be mocked, reducing spo-

radic test failures and improving test reliability. Continuous Integration enables the automatic and regular execution of the test suite, ensuring that recent code changes do not introduce regressions and supporting a robust and safe development process.

- Performance locality. Keyed access, indexing, and time-series-aware iterators enable efficient lookups and cache-friendly traversal, improving throughput at scale.
- Concurrency and parallelism. Independent *Scenario* instances enable parallel processing, while immutable snapshots allow safe data sharing across threads and processes.
- Interoperability across languages. A well-defined and stable data model schema facilitates language bindings (e.g., C++/Python) and integration with external tools, as data contracts are explicit and independent of the transport layer. This enables seamless reading, writing, and manipulation of the data model from Python, paving the way for developing Python-based extensions and application-specific modules without modifying the underlying C++ codebase (see Section 2.2).
- Data quality enforcement. Referential integrity and schema validation catch inconsistencies early, improving robustness in long-running operations.

The data model is conceptually similar to a relational database, in which entities are linked through unique identifiers (keys). In this framework, satellites and ground stations are associated with instruments (e.g., GNSS receivers or Laser Retro- Reflectors – LRR), which in turn are associated with antennas and oscillators. This structure is illustrated in Fig. 1, which shows an example configuration for a satellite (left) and a ground station (right). Instruments (grey boxes) are represented as distinct entities and may be connected to one or more antennas (top of the figure). In addition, instruments such as GNSS or DORIS receivers can be linked to a reference oscillator (in the middle of the figure), whereas passive instruments such as LRRs are not associated with an oscillator (instruments at the bottom of the figure). This explicit representation of physical components and their relationships enable flexible configuration handling and supports consistent multi-technique processing.

Internally, the data model is implemented using C++ Standard Template Library (STL) containers keyed by unique identifiers (IDs), together with an index pointing to the most recently accessed element—an optimization that is particularly effective for time-series data. These containers, referred to as tables, are managed by a central C++ class called Scenario. This class owns all data tables and provides controlled access to their contents, while also handling object lifecycles and memory management. Notably, Scenario is not implemented as a singleton, allowing multiple independent scenarios to coexist. This design enables parallel processing at the cost of limited data duplication.

Within *FocusPOD*, two conceptual types of tables are defined:

- Logical Entities – These represent real-world physical components or systems that have a persistent existence, such as satellites, ground stations, instruments, antennas, and oscillators. These entities correspond to the elements shown in Fig. 1 and to the red boxes in Fig. 2.

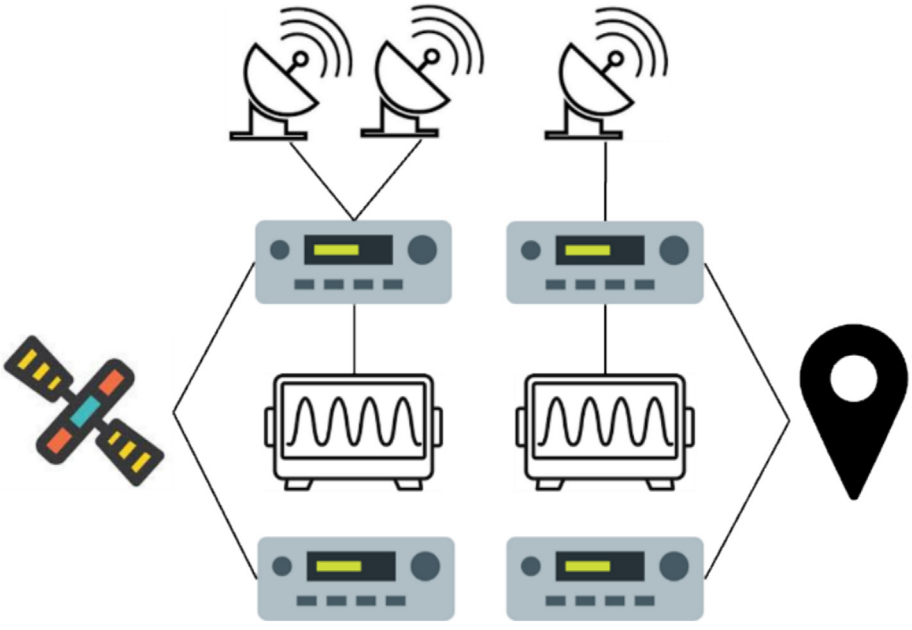


Fig. 1. Example of relationships between physical entities in the *FocusPOD* data model. The figure illustrates a satellite (left) and a ground station (right), each associated with multiple instruments (rectangular grey boxes). Instruments may be connected to one or more antennas (upper) and to a reference oscillator (middle). This example highlights how physical components are represented as distinct entities and linked through explicit relationships.

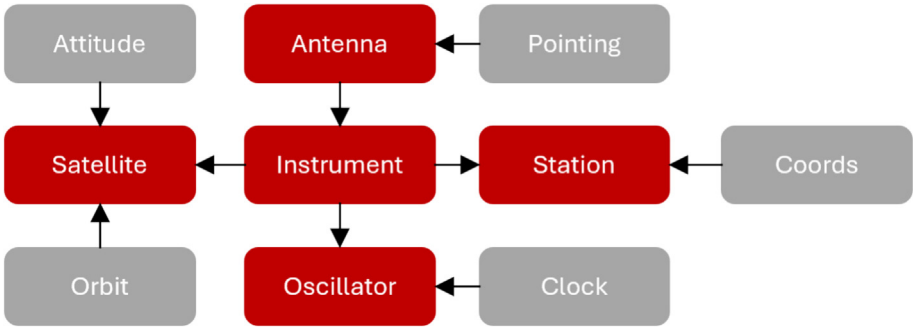


Fig. 2. Example of data model relationships for a subset of elements. Red boxes denote logical entities representing real-world physical components, while grey boxes denote physical data describing time-varying information.

- Physical Data – These represent time-varying quantities describing the physical state or behaviour of logical entities, such as orbits, clocks, attitudes, antenna pointing, or station coordinates. These are illustrated by the grey boxes in Fig. 2.

Physical data may be associated with a logical entity (e.g., an orbit linked to a satellite) or may exist independently (e.g., Earth Orientation Parameters).

2.1.4. Multiple use cases

Another key principle of *FocusPOD* is enabling flexibility in how the library is integrated and used. The architecture supports a wide range of usage patterns, from standalone executables to distributed services and interactive workflows. Concretely, examples of how *FocusPOD* can be used include:

- End-to-end Instrument Processing Facility (IPF) in a Payload Data Ground Segment (PDGS) (C++ or Python). A single process performs the entire precise orbit computation, from reading initial inputs to producing final products, without intermediate I/O, except at the start and end. This is how it is used on the CPOD Service (Section 3) and the reprocessing presented in Section 4.
- Distributed, service-oriented systems. Independent components (e.g., propagation, estimation, quality control) communicate over well-defined interfaces (e.g., Representational State Transfer (REST) API), allowing horizontal scaling and fault isolation.
- Notebook-driven R&D and teaching. Use Python bindings in Jupyter Notebooks for rapid prototyping, visualization, and reproducible experiments with the same core algorithms.

This versatility is enabled by clean C++/Python APIs, a stable data model, and transport-agnostic I/O, allowing *FocusPOD* to adapt to diverse development workflows and operational environments.

2.2. Python extensibility

Inspired by modern programming practices, *FocusPOD* has been enhanced to interface with Python. The interface between the C++ core of *FocusPOD* and Python is implemented using the pybind11 library, which provides lightweight and efficient bindings without introducing additional runtime dependencies. Through pybind11, C++ classes and functions are exposed directly to Python, enabling both execution orchestration and direct interaction with the underlying data model.

FocusPOD supports three complementary modes of interaction with Python. First, it allows executing the algorithms of the C++ library directly from Python. This enables orchestration of the execution flow through Python scripts rather than compiled C++ programs, offering greater flexibility and accessibility. An example of this capability is described in Section 3.2. Second, it supports direct manipulation of the data model from Python. This includes reading and writing data, as well as applying algorithms to the data. Such functionality enables the development of specialized providers (data readers and writers) or algorithms that are external to the C++ core but tailored for specific applications or clients. For instance, a Python-based provider can handle a proprietary file format, allowing the client to develop and maintain it independently while remaining fully compatible with the *FocusPOD* framework. Third, and still under active development, *FocusPOD* will allow C++ algorithms to be extended with Python modules. For example, an orbital perturbation such as a new solar radiation pressure model can be implemented in Python and seamlessly integrated into the C++ propagator. This approach accelerates prototyping and testing, while empowering users to extend existing functionalities without modifying the C++ core.

Fig. 3 illustrates this architecture: components shown in red correspond to the C++ core, including the data model, while the black and purple elements represent Python components used to orchestrate execution. The purple blocks denote user-specific extensions developed for particular applications; these include *providers* (for reading and writing proprietary formats to and from the data model), algorithms that directly manipulate the data model, and Python-based algorithms that extend the C++ base classes and integrate with the C++ code.

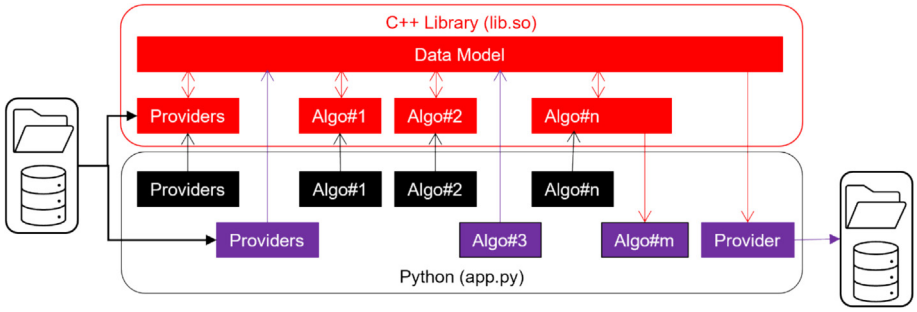


Fig. 3. Schematic representation of the use of the *FocusPOD* library. Core C++ components are shown in red, Python orchestration and extensions in black and purple. Providers implement input and output interfaces, enabling data to be read from or written to files or databases. Algorithms may be implemented either in C++ (red) or in Python (purple), while execution and workflow orchestration are handled through Python.

2.3. External dependencies

The choice of programming languages (C++ and Python), together with principles of economy and efficiency, motivated the use of external libraries in *FocusPOD* in areas where mature solutions are available. These include not only scientific libraries, but also general-purpose libraries for common tasks such as logging, parsing, and language binding. Table 2 lists the external libraries used by *FocusPOD*.

2.4. Algorithms and models

FocusPOD provides a broad and advanced set of capabilities supporting astrodynamics, POD and geodetic applications. Its functionality spans data handling, physical modelling, simulation, and estimation, supporting interoperability across multiple GNSS constellations and space-geodetic techniques within a unified framework.

Data Handling and Standards: All input/output data in *FocusPOD* are managed through a central *Scenario* data container (see Section 2.1.3). A dedicated module handles standard formats for reading and writing external data. *FocusPOD* supports all major astrodynamics and geodesy data standards, including the Receiver Independent Exchange Format (RINEX) v3/v4 (Gurtner and Estey, 2007; Gini and Hauschild, 2025) and RINEX DORIS v3.0 (CNES, 2022), SP3-c/d (Hilla, 2010, 2016), RINEX Clock (Ray and Gurtner, 2010; Ray et al., 2017), ILRS CRD laser ranging data (v1/v2) (Ricklefs and Moore, 2009; ILRS, 2019) and CPF prediction format (ILRS, 2018), VLBI vgosDB observation files (Gipson, 2021), SINEX solutions for station coordinates, Earth Orientation Parameters (EOPs) and correlation matrices (IERS, 2006) as well as biases solutions (Schaer, 2018), and ORBEX (Loyer et al., 2019) for orbit and attitude.

Reference Frames and Time Systems: High-precision reference-frame transformations follow the International Earth Rotation Service (IERS) Conventions 2010 (Petit and Luzum, 2010), including conversions between Earth-fixed (International Terrestrial Reference Frame – ITRF) and inertial (Geocentric Celestial Reference Frame – GCRF) frames, with the appropriate intermediate frames (Celestial Intermediate Reference Frame – CIRF, Terrestrial Intermediate Reference Frame – TIRF). IERS Conventions 1996 (McCarthy, 1996) are also implemented for specific applications, for instance, the modelling of the Sentinel-1 attitude requires their use. For specific applica-

Table 2
List of external SW dependences of *FocusPOD*.

Name	Description
SOFA	Standard models used in fundamental astronomy, based on IERS Conventions. https://www.iausofa.org
SGP4	SGP4 propagator for TLEs. https://celestrak.com/software/vallado-sw.php
IERS2010	Set of routines that extend the IERS 2010 conventions. https://github.com/xanthospap/iers2010/
Calceph	Access the binary planetary ephemeris files, such INPOPxx, JPL DExxx and SPICE ephemeris files. https://www.imcce.fr/inpop/calceph
NRLMSISE-00	NRLMSIS 00 Atmosphere Model. https://www.brodo.de/space/nrlmsise/
Eigen	Template library for linear algebra: matrices, vectors, numerical solvers, and related algorithms. https://eigen.tuxfamily.org
YAMLCpp	A YAML parser and emitter. https://github.com/jbeder/yaml-cpp
FMT	Modern formatting library. https://github.com/fmtlib/fmt
SPDLog	Fast C++ logging library. https://github.com/gabime/spdlog
Xerces-c	Validating XML parser. https://xerces.apache.org/xerces-c
bxzstr	Access to compressed streams. https://github.com/tmaklin/bxzstr
Thread-pool	Thread Pool for parallel processing. https://github.com/bshoshany/thread-pool
Pybind11	Python binding library. https://github.com/pybind/pybind11

tions such as Two-Line Elements (TLE) propagation, the True Equator Mean Equinox (TEME) frame is also supported. *FocusPOD* handles multiple time scales such as International Atomic Time (TAI), which is used internally, Coordinated Universal Time (UTC) and GNSS system times (e.g., GPS and Galileo).

The implementation has been generalized to support both rotations (axes) and translations (centres), so transformations are not Earth-centric:

- Centre-aware frame model. A *frames* module exposes explicit axes and centre components enabling state-vector conversions that account for both orientation changes and origin shifts.
- Composable transformations with path finding. Conversion paths are determined automatically by chaining basic rotations and translations, replacing hard-coded trees; this simplifies adding terrestrial or planetary frames and avoids Earth-specific assumptions.
- Ephemeris-based translations. When involving celestial bodies, centre-to-centre translations rely on high-precision planetary ephemerides, enabling conversions among any defined celestial bodies (e.g., Earth, Moon, Sun).
- Performance and accuracy safeguards. Rotation matrices, angular rates, and translation vectors are cached for efficiency.

Thanks to a body-agnostic design with centre-aware frames, composable transformations, and ephemeris-based translations, interplanetary and local (station or spacecraft) conversions require only a body's axes rotation and centre translation models. This keeps Earth–Moon–Mars (and beyond) transformations seamless while preserving typical Earth-based workflows.

2.4.1. Physical modelling and simulation

FocusPOD includes a high-fidelity orbit propagation engine supporting both numerical integration and analytical propagation. For numerical integration, users can choose from fixed-step stepper (e.g. Runge–Kutta and multi-step Adams-Bashforth-Moulton methods) or adaptive step algorithms (e.g. Runge–Kutta–Fehlberg 7/8) (Montenbruck and Gill, 2012) to propagate the spacecraft state. The dynamic models cover all relevant forces and perturbations needed for state-of-the art POD accuracy: a detailed geopotential with Earth gravity field models (including time-varying spherical harmonic terms) and solid Earth & ocean tide perturbations, atmospheric gravity, third-body Solar System gravitational attractions (Sun, Moon, etc.), relativistic accelerations, direct solar radiation pressure (with options for fixed cross-sectional area or detailed satellite macro-model), Earth albedo and infrared radiation pressure, atmospheric drag (including use of empirical atmospheric models such as MSISE-00 (Picone et al., 2002) and optional spacecraft macro-model for variable cross-section), empirical acceleration models (e.g. once-per-revolution periodic terms and empirical CODE orbit model ECOM, ECOM-2 (Arnold et al., 2015), for solar radiation pressure) and antenna thrust. Spacecraft manoeuvres are supported in both impulsive and continuous low-thrust form. For comparison and fast computations, simpler analytical propagators are available as well: Keplerian two-body motion and the Simplified General Perturbations-4 (SGP4/SDP4) (Vallado, 2013) models for objects with TLE data. The software can also propagate covariance matrices alongside the state (linearized covariance propagation) for uncertainty analysis.

FocusPOD provides tools for geometrical event detection (e.g., eclipses, passages through the ascending or descending node, and visibility of a satellite from a ground station, etc.) as well as simulation of mission geometry. An *Events* module computes events via root-finding algorithms. The library also supports high-order interpolation algorithms to efficiently resample or synchronize time series of orbits, attitudes, clock offsets, EOP, etc., without loss of accuracy. In addition, *merging algorithms* are available to combine or replace trajectory and timing data from different sources while preserving consistency. For example, a user can seamlessly merge predicted and precise ephemerides (overwriting predicted segments with higher accuracy computed orbits) or align clock solutions from different sources to a common reference timescale. This facilitates long-arc orbit concatenation and multi-mission data fusion. Finally, *FocusPOD* includes an observation simulation capability to generate synthetic tracking data given a truth orbit and models (useful for testing and mission design).

2.4.2. Observation modelling and tracking techniques

A major strength of *FocusPOD* is its ability to handle all four space geodetic tracking techniques within a unified framework. The library supports GNSS, SLR, DORIS and VLBI observations through a shared observation-processing layer that ensures consistent modelling across techniques.

2.4.3. GNSS

The software can ingest dual-frequency code, carrier-phase and Doppler measurements from multiple constellations (validated so far with GPS and Galileo). A comprehensive set of observation models commonly used in POD is implemented: receiver and satellite clock offsets (with relativistic corrections for clock rates and signal propagation delay), tropospheric delay corrections (currently supporting Mendes and Pavlis (2004), Niell (1996), and Saastamoinen (1972) for zenith delay plus mapping functions; the framework is designed to accommodate more advanced tropospheric models in future developments), ionospheric delays (handled via dual-frequency combinations or empirical models like Galileo's NeQuick (EC, 2016) for single-frequency data), carrier-phase wind-up effects due to satellite/receiver attitude, phase centre offsets and variations (PCO/PCV) for antennas, and various

instrumental biases. Both absolute and differential biases are accounted for in the observation model. The GNSS model also includes carrier-phase ambiguities as estimable parameters, enabling precise positioning when fixed to integers.

2.4.4. SLR

FocusPOD models two-way laser range observations using the standard ILRS/IERS-conformant formulation, accounting for relativistic effects, station motion, and atmospheric propagation corrections. Atmospheric refraction is corrected using optical delay models (e.g., Mendes and Pavlis, 2004), and both station-dependent and satellite-dependent range biases can be estimated or corrected. Variations of the satellite optical centre can be corrected but are not currently estimated.

2.4.5. DORIS

The software reads DORIS RINEX phase and power data, performs the ionosphere-free combination, reconstructs Doppler counts (without ionosphere), and converts them to range-differences on the primary frequency, accounting for the on-board receiver clock estimates. It applies the same force and observation models as for GNSS and SLR, including tropospheric corrections. A per-pass and per-station bias is estimated for each ground beacon to compensate frequency errors.

2.4.6. VLBI

FocusPOD implements the IERS conventional model for group delay observables, processing vgosDB sessions with ICRF3 source catalogues. The modelling includes geometric delays, ionospheric and tropospheric corrections, and station-specific effects such as clock offsets, antenna deformation, and cable delays. Both clock offsets and tropospheric parameters are treated as piecewise linear polynomials. This modelling ensures full consistency with other techniques, enabling rigorous multi-technique geodetic combinations and future Terrestrial Reference Frame derivations.

While *FocusPOD* includes support for DORIS, SLR, and VLBI observation processing, a full scientific validation of these techniques, as well as of fully combined multi-technique orbit solutions, is currently under validation and therefore it is outside the scope of the present study. Accordingly, this article demonstrates *FocusPOD* exclusively in the context of GNSS-based LEO POD.

2.4.7. Estimation algorithms and performance

The core estimation engine in *FocusPOD* uses both batch and sequential filtering methods to solve for spacecraft orbits and other parameters. A Weighted Batch Least-Squares (BLSQ) estimator iteratively adjusts the parameters to minimize observation residuals, applying configurable weighting schemes for different data types. In parallel, an EKF is available for sequential or real-time estimation needs. The estimation framework is flexible and multi-parameter; it supports simultaneous estimation of orbital state vectors and a wide range of ancillary parameters like satellite dynamical parameters (e.g. empirical accelerations, solar radiation pressure scaling, drag coefficients), measurement biases (e.g. GNSS inter-system biases, DORIS station biases, SLR station and satellite biases), phase ambiguities as constant per pass, receiver and satellite clock biases (estimated epoch-by-epoch or as continuous polynomials), Earth Orientation Parameters (polar motion, UT1 and celestial pole offsets linear models, including Length of Day – LOD) and station coordinates, and environmental parameters like tropospheric zenith delays for each station (with optional estimation of mapping function). The software natively supports snapshot estimation for high-rate parameters like clocks, which are represented as white-noise parameters in an EKF or as independent parameters in each epoch of a BLSQ solution. Integer Ambiguity Resolution (IAR) algorithms are integrated to fix GNSS carrier-phase ambiguities to their integer values when the solution quality allows, improving positioning precision to the sub-centimetre level. This IAR capability has been demonstrated in single-receiver POD and has been extended to network scenarios. Additionally, the estimator supports (i) a priori (initial-value and sigma) constraints formulated as pseudo-observations, (ii) bound (box) inequality constraints on parameters, and (iii) minimum-datum constraints for GNSS network processing, enforced as Helmert-type conditions: no-net-rotation, no-net-translation, and no-net-scale. Relative time-dependent constraints enforcing parameter continuity are not yet implemented.

The estimation framework of *FocusPOD* is designed to support multi-satellite and multi-station solutions, including the simultaneous estimation of station coordinates, EOPs, satellite orbits, clock biases and measurement biases from GNSS, SLR, DORIS and VLBI observations within a unified adjustment. While the modelling and estimation infrastructure for these capabilities is implemented (Table 3), the present article demonstrates and fully validates *FocusPOD* exclusively in the context of GNSS-based LEO POD. Full scientific validation of multi-station GNSS network solutions, DORIS- and SLR-based LEO POD, and combined multi-technique geodetic solutions is currently ongoing and will be presented in future publications.

Although *FocusPOD* places strong emphasis on software architecture, these design choices have direct scientific implications. The unified relational data model ensures that all observations, parameters, and models are handled consistently across techniques, reducing the risk of hidden inconsistencies and unintended correlations. The strict separation between data and algorithms enables identical estimation models to be reused across POD, simulation, and reprocessing scenarios, improving reproducibility. The ability to execute complete estimation workflows without mandatory intermediate I/O not only improves computational performance but also preserves numerical precision

Table 3
Measurements and models available at *FocusPOD*.

Element	Description
Measurement type	
PVT	Position, Velocity, Time (as estimated onboard by a GNSS receiver)
GNSS	Code, Phase, Doppler, CN0
DORIS	Phase, Power
SLR	Ranging
VLBI	Group Delay
Optical	Angles
RF	Ranging, Angles, Doppler
TDoA	Time difference of arrival
Frames	TRF (ITRF), CRF (GCRF), Earth orientation and frame transformations
Polar motion and UT1	IERS finals2000A.data (Petit and Luzum, 2010)
Pole model	IERS 2010 Conventions (Petit and Luzum, 2010)
Precession/Nutation	IERS 2010 Conventions
Geocenter motion	ITRF20 model (Altamimi et al., 2023)
Gravity	
Gravity field	Support different geopotentials (e.g., COST-G (Peter et al., 2022), EIGEN (Lemoine et al., 2019, 2023), EGM (Lemoine et al., 1998))
Solid Earth tides	IERS 2010 Conventions
Ocean tides	FES2014/22 (Lyard et al., 2021, CNES, 2024)
Atmospheric gravity	GFZ AOD L1B RL06/7 (100 × 100) (Dobslaw et al., 2017)
Atmospheric tides	GFZ AOD L1B RL06/7 (100 × 100)
Earth pole tide	IERS 2010 Conventions
Ocean pole tide	IERS 2010 Conventions
Third bodies	Sun, Moon, Planets DE-421 (Folkner et al., 2009)
Relativity	IERS 2010 Conventions
Surface forces and empirical	
Solar Radiation Pressure	Milani et al. (1987). Macro model (with instantaneous re-radiation)
Earth Radiation Pressure	Knocke et al. (1988) formulation with CERES Earth radiation data (Priestley et al., 2011)
Atmospheric density model	msise00 (Picone et al., 2002); NOAA, GFZ indexes (Matzka et al., 2021)
Empirical accelerations	Cycle-per-revolution (CPR) Empirical CODE Orbit Model (ECOM, ECOM-2) (Arnold et al., 2015)
Antenna thrust	Steigenberger et al. (2018)
Ionosphere	NeQuick-G (EC, 2016)
Troposphere	Mendes and Pavlis (2004) Saastamoinen (1972) Niell (1996)
Station deformation	Solid Earth tides, ocean tidal loading, atmospheric loading, pole tide Post-seismic deformation (Altamimi et al., 2023)
Integer Ambiguity Resolution	Montenbruck et al. (2018)
GNSS Phase wind-up	Wu et al. (1993)

and facilitates large-scale reprocessing. Together, these aspects enable complex estimation configurations—such as multi-technique processing, high-rate parameter estimation, and long-time series analyses—that could be difficult to manage reliably in more fragmented software architectures.

FocusPOD is currently distributed under a commercial license and is used within GMV-led operational and institutional projects. It is not released as open-source software. Access for external partners is granted within specific contractual or collaborative frameworks. The possibility of extending the licensing model to scientific institutions is under evaluation, with the objective of enabling research use while maintaining long-term sustainability of the software. A dual-use licensing scheme supporting both commercial and scientific applications is therefore being considered for future phases.

3. Copernicus POD service

3.1. Introduction

This section shows the first use of the *FocusPOD* library for GNSS LEO POD in an operational environment, the CPOD Service (Fernández et al., 2024), which is responsible for the operational generation of precise orbital products and auxiliary files of the Copernicus Sentinel satellites. Table 4 summarizes the operational periods, orbital characteristics (altitude and inclination), and onboard POD instruments of the Sentinel missions currently processed by the CPOD Service.

Table 5 provides the main characteristics of the CPOD orbital products, including their production latency, the orbit accuracy requirements defined in the corresponding Mission Requirement Documents (MRDs), and the

Table 4
Summary of Copernicus Sentinel missions.

Mission	Satellite	Launch/Decom.	Altitude (km)	Inclination (deg)	POD instruments
Sentinel-1	S1A	2014-04-03	693	98.18	S1[AB]: GPS
	S1B	2016-04-25/2022-08-03			S1C: GPS + GAL
	S1C	2024-12-05			
Sentinel-2	S2A	2015-06-23	786	98.62	S2[AB]: GPS
	S2B	2017-03-07			S2C: GPS + GAL
	S2C	2024-09-05			
Sentinel-3	S3A	2016-02-16	815	98.65	GPS, SLR, DORIS
	S3B	2018-04-25			
Sentinel-6	S6A	2020-11-21	1336	66.0	GPS + GAL, SLR, DORIS

current Service Level Agreement (SLA) (i.e., the formal contract defining the expected service performance, quality, and responsibilities between a provider and a client) accuracy requirements, evaluated on an annual basis with accuracy values for low and high solar activity.

3.2. GNSS LEO POD orchestration

The architecture of *FocusPOD* allows to implement multiple ways to perform a GNSS-based LEO POD. Fig. 4 presents the approach currently used for the operations of the CPOD Service.

The GNSS LEO POD tool is orchestrated in Python 3 and contains the following steps:

- Tool initialization: It includes the minimum steps to initialise the tool, including setting the logger level and the loading of the user configuration file (YAML format).
- Process input data: It builds the *Scenario* where the program data resides, and populates it reading the inputs files (e.g., models, GNSS products and observations, quaternions, etc.)
- Observation preprocessing: Once all input data is loaded into the *Scenario*, the preprocessing carries out several modifications to the observations. It firstly applies code and phase biases if available, followed by computing the ionosphere-free (IF) linear combination (LC) of the selected signals. After that, it removes outliers using several criteria and identifies passes. Finally, it computes a kinematic solution using the IF LC code observations and the Bancroft algorithm (Bancroft, 1985). As a result, a large number of state-vectors are computed and loaded into the *Scenario*. The use of the Bancroft algorithm at this stage enables the generation of an a priori orbit solution without relying on external sources, such as on-board navigation solutions or the orbit from the previous day, which may be erroneous or unavailable. Furthermore, this approach computes state vectors independently whenever sufficient observations are available, allowing data gaps to be skipped naturally without compromising the preprocessing.
- POD least square estimation (fitting to kinematic): A first POD estimation is performed using a kinematic solution, namely the XYZ position estimates obtained from the Bancroft algorithm during preprocessing. This step is called in our chain Single Point Positioning (SPP) as it is based on the XYZ position estimates. For this fitting, the a priori state vector is selected from the Bancroft solutions as the one with the smallest RMS, ensuring that the most reliable initial state is used. Additional safeguards are applied to avoid selecting state vectors too close to the beginning or end of the processing interval, which would otherwise require short propagation times and could degrade the robustness of the initialization. This initial POD step produces a smooth

Table 5
Summary of CPOD orbital products characteristics.

Product	Latency	MRD – orbits	CPOD SLA global req.
Predicted (PRE)	S1: 30 min	S1: 100 cm (3D RMS)	N/A
Near-Real Time (NRT)	S1: 30 min	S1: 10 cm (3D RMS)	S1: 3.9–4.7 cm (3D RMS)
	S2: 30 min	S2: 300 cm (3D RMS)	S2: 3.9–4.8 cm (3D RMS)
	S3: 10 min	S3: 10 cm (rad RMS)	S3: 2.3–2.1 cm (rad RMS)
	S6: 10 min	S6: 5 cm (rad RMS)	S6: 1.7–1.7 cm (rad RMS)
Short Time Critical (STC)	30 h	4 cm (rad RMS)	S1: 2.9–3.1 cm (3D RMS)
			S3: 1.1–1.2 cm (rad RMS)
Non Time Critical (NTC)	S1: 21 days	S1: 5 cm (3D RMS)	S1: 0.8–0.9 cm (3D RMS)
	S3: 25 days	S3: 3 cm (rad RMS)	S3: 0.5–0.5 cm (rad RMS)

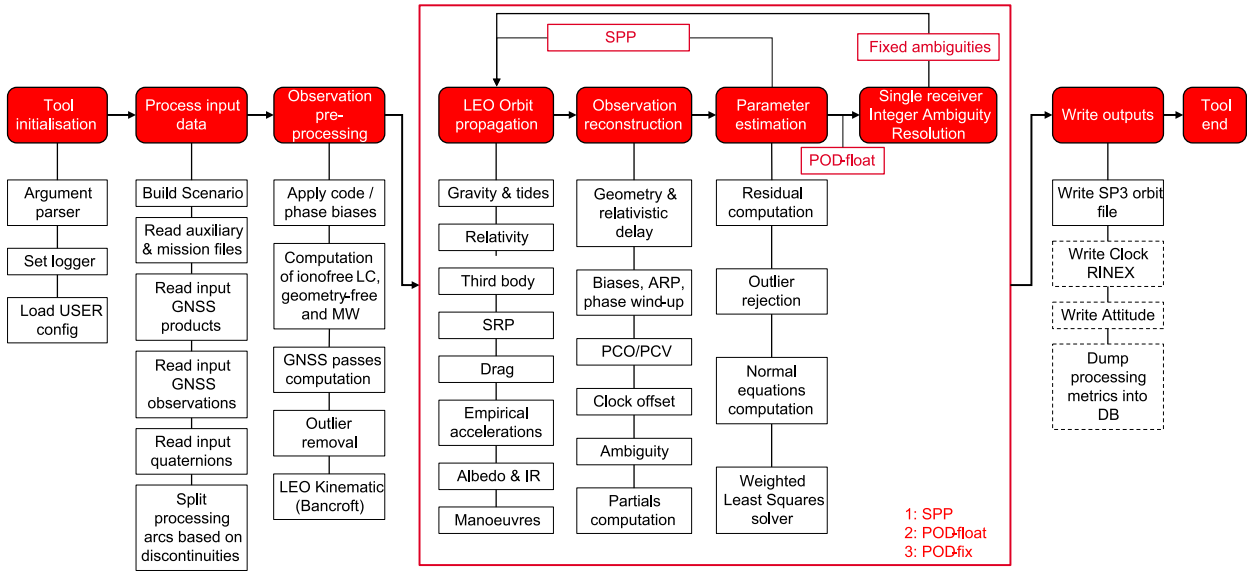


Fig. 4. High-level overview of the GNSS LEO POD processing chain implemented with *FocusPOD* in the CPOD Service. Red boxes denote the conceptual envelope of the main processing steps, while white boxes indicate the corresponding sub-steps. The POD block (big square in red) is executed three times in sequence, corresponding to the SPP initialization, the POD solution with float ambiguities, and the POD solution with fixed ambiguities.

orbit that filters the noise present in the kinematic solution and bridges potential data gaps. This step allows to estimate a decimetre level accuracy of the orbit that is used as initial solution for the subsequent POD step.

- POD least square estimation (float ambiguities): The second POD estimation uses IF LC code and carrier-phase observations from one or more GNSS constellations, together with a high-fidelity orbital model, while estimating carrier-phase ambiguities as float parameters. The a priori state vector is initialized from the solution obtained in the previous step, selected at the midpoint of the processing interval.
- Single receiver Integer Ambiguity Resolution: This step fixes the ambiguities using the method described by Montenbruck et al. (2018).
- POD least square estimation (fixed ambiguities): The third POD estimation is equal to the second POD except that the ambiguities are fixed to the value provided in the previous step.
- Write outputs: This step generates the products required, that can include an orbit file in SP3 or another format, clock RINEX, attitude, etc. Additionally, the processing metrics can be dumped into a YAML file or a database like PostgreSQL.
- Tool end: the Scenario is deleted, and the execution of the tool is ended.

One of the main advantages of *FocusPOD* is that all processing steps are executed within a single run. Owing to its architecture, and in particular to the underlying data model, the entire workflow can be orchestrated without the need to write or read intermediate files, although intermediate states can be explicitly persisted when required for debugging or recovery. This not only reduces processing time but also preserves the precision of floating-point numbers throughout the computation.

Table 6 summarizes the main processing evolutions implemented in the CPOD Service since 2021. Notably, these include the migration to *FocusPOD* in January 2023, the adoption of the COST-G gravity field in July 2023, and the transition to the IGS20 reference frame in November 2022.

3.3. Results

Fig. 5 illustrates the orbit determination performance of the CPOD Service since 2021, highlighting the *neutral impact* on precision after the transition to *FocusPOD* and the *positive effect* of adopting the COST-G geopotential model. Each panel corresponds to one representative satellite from each Sentinel family. For Sentinel-1A and Sentinel-2B, the metric is the 3D RMS, while for Sentinel-3A and Sentinel-6A it is the radial RMS. The differences are computed with respect to the combined solution (see Fernández et al., 2022; Fernández et al., 2024 for details on the orbit combination methodology). Each plot displays three types of products: NRT (in red), typically computed in under five minutes; STC (in purple), produced within the following day; and NTC (in black), delivered approximately three weeks after data acquisition. The 1-σ (68th percentile) values of each metric (3D and radial RMS) are shown to demonstrate consistency across missions and the remarkable performance. For Sentinel-1A and Sentinel-2B, the 3D RMS (1-σ) values are approximately 3.1 cm and 3.0 cm for the NRT products, decreas-

Table 6
Main processing evolutions in the CPOD Service since 2021.

Parameter	v1.10.0	v2.0.0	v3.0.0	v3.3.0
Date	Feb 2021	Nov 2022	Jan 2023	July 2023
POD SW	NAPEOS		*FocusPOD*	
IERS Conventions	IERS 2010 (Petit and Luzum, 2010)			
Gravity Field	EIGEN.GRGS.RL04 TVG (120 × 120) (Lemoine et al., 2019)			COST-G (90 × 90) (Peter et al., 2022)
Ocean Tides	FES2014 (100 × 100, 142 tidal constituents) (Lyard et al., 2021)			
Atmospheric gravity	GFZ AOD L1B RL06 (100 × 100) (Dobslaw et al., 2017)			
Number of drag parameters	NRT: 10 estimated per 24 h			
	STC/NTC: 1 estimated for S123; 1 fixed for S6A to 1.			
Solar radiation pressure	NRT: 1 estimated			
	STC/NTC: 1 fixed (S126 to 1.0, S3A to 0.97 and S3B to 0.96)			
Empirical accelerations (CPR)	NRT: 2 estimated per 24 h; sine/cosine in along-track/cross-track			
	STC/NTC: 16 (S123) and 8 (S6) per 32 h; constant/sine/cosine in along-track/cross-track			
Phase ambiguity	NRT/STC: Float			
	NTC: Integer			
Frame	IGb14 (Rebischung, 2020)	IGS20 (Villiger, 2022)		
GNSS products	NRT/STC: External GNSS Provided by GMV GSharp (GMV, 2023)			
	NTC: CODE finals			
Arc length	NRT: 24 h			
	STC/NTC: 32 h			

ing to about 1.9 cm and 1.7 cm for the STC products and to around 0.5 cm for the NTC products. For Sentinel-3A and Sentinel-6A, the corresponding radial RMS values are approximately 1.4 cm and 1.1 cm for NRT, 0.6 cm and 0.7 cm for STC, and about 0.3 cm for NTC.

These performance differences are primarily driven by the increasing latency of the products, which enables both improved data usage and more refined modelling strategies. The NRT products are generated shortly after data acquisition and typically cover only one or two orbital revolutions at the edge of the 24 h processing window. As a result, the comparison interval often corresponds to a worst-case scenario in terms of orbit geometry and data completeness, leading to larger dispersion in the NRT statistics.

In contrast, the STC and NTC products benefit from longer data availability and more favourable arc geometry, even though the comparison itself is performed over a 24 h interval. The additional processing time also allows the use of a more reduced-dynamic parameterization, with a larger number of estimated empirical acceleration parameters (up to 8/16 CPR terms for STC and NTC, compared to only 2 for NRT), which significantly improves the modelling of unmodelled forces.

Finally, the NTC products further benefit from the availability of high-quality GNSS code and phase bias products, enabling carrier-phase ambiguity fixing. This step provides a substantial improvement in orbit precision, explaining the additional performance gain observed for the NTC solutions.

Fig. 6 shows the yearly evolution of the 1-σ (68th percentile) values of the 3D or radial RMS for the NRT (left) and NTC (right) products. For Sentinel-1A and Sentinel-2B, the metric shown is the 3D RMS, while for Sentinel-3A and Sentinel-6A, the radial RMS is used, as required by altimetry missions. In the left plot (NRT), the effect of increasing solar activity between the low activity period around 2021 and the high-activity conditions observed in 2023–2024 is evident, followed by a slight improvement in 2025 for Sentinel-1A and Sentinel-2B. This variability is not visible in the radial RMS of Sentinel-3A and Sentinel-6A probably due to its higher altitude. In the right plot (NTC), the solar activity effect is also observed for Sentinel-1A and Sentinel-2B, but the most notable improvement occurs between 2023 and 2024, coinciding with the implementation of the COST-G geopotential model.

Fig. 7 presents the impact of using *FocusPOD* on processing timeliness since January 2023. The left panel shows the processing time of NRT products for Sentinel-3A and Sentinel-6A, where a reduction of approximately two minutes is clearly visible after the introduction of *FocusPOD*. The right panel shows the average yearly processing times, demonstrating that NRT products can now be routinely generated in less than five minutes, typically around 3.5 min. The dispersion observed in the left panel is mainly due to the 1-min resolution used to launch the processing,

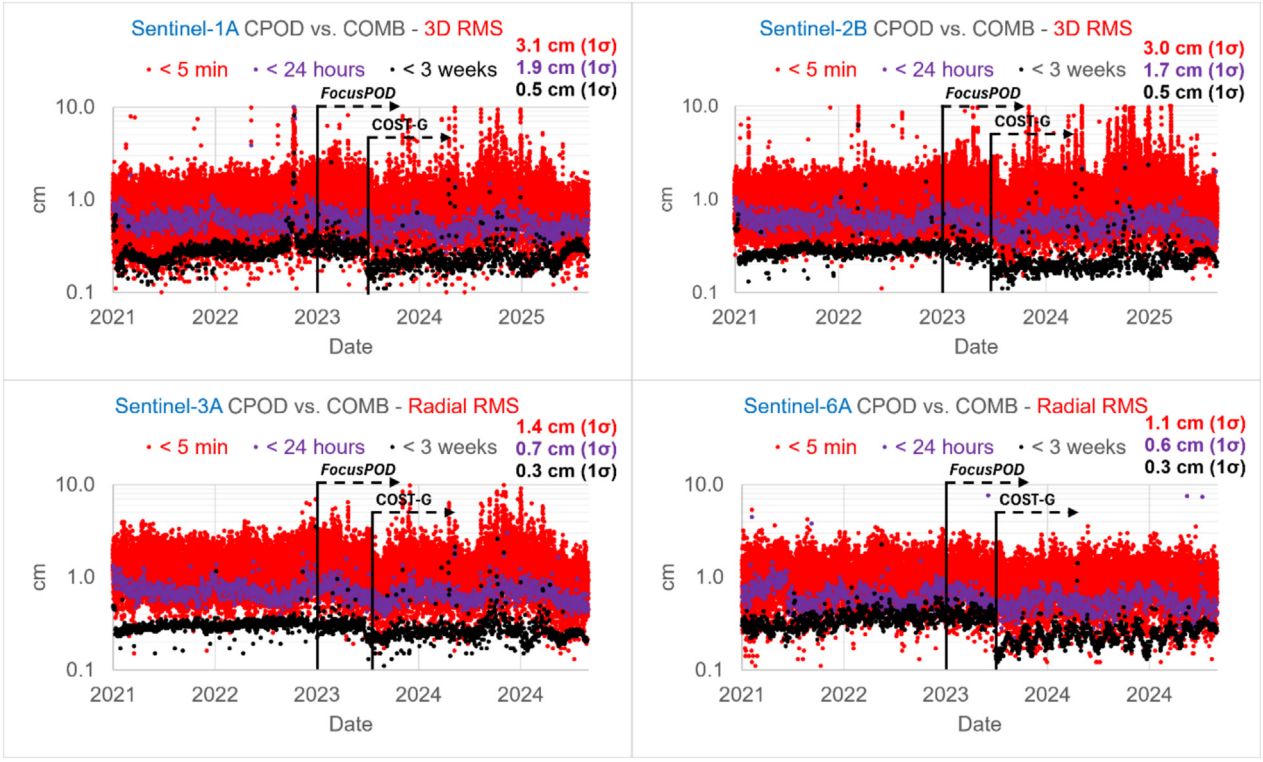


Fig. 5. CPOD orbital products with respect to the combined solution (COMB) for the period 2021–2025 and satellites S1A, S2B, S3A, and S6A. Each panel shows the 3D or radial RMS for products of different timeliness: NRT (red), STC (purple), and NTC (black). The 1-σ (68th-percentile) values for each timeliness category are indicated within each panel. The figure also shows when *FocusPOD* and COST-G were introduced.

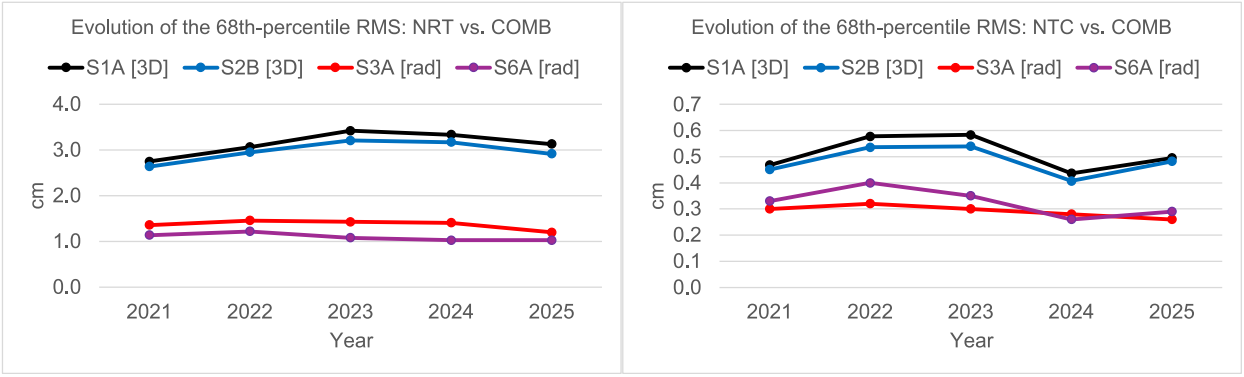


Fig. 6. Yearly evolution of the 68th-percentile (1-σ) RMS values of the orbit comparison with respect to the combined solution (cm). Left: NRT products; right: NTC products. The comparison metric is the 3D RMS for S1A and S2B, and the radial RMS for S3A and S6A, in accordance with the mission-specific requirements.

implying that, in practice, NRT products can potentially be generated in about three minutes once the input data are available.

An additional measure of the precision of *FocusPOD* is obtained through comparisons with the solutions contributed by the CPOD Quality Working Group (QWG) relative to the combined orbit solution. Fig. 8 shows the average daily 3D RMS of the differences between the orbits of S1A, S2B, S3A, and S6A computed by different CPOD QWG centres using distinct POD software and processing strategies, and the combined solution, which is derived as a weighted mean of these individual solutions. Table 7 summarizes the POD software used by each analysis centre, the corresponding institutions, the GNSS products and observations employed, and additional relevant comments. The results show that the level of agreement among most solutions is on the order of 5–10 mm in 3D RMS.

4. Reprocessing of sentinels missions

4.1. Introduction

One of the recurrent tasks in space geodesy is the execution of large-scale reprocessing campaigns to generate con-

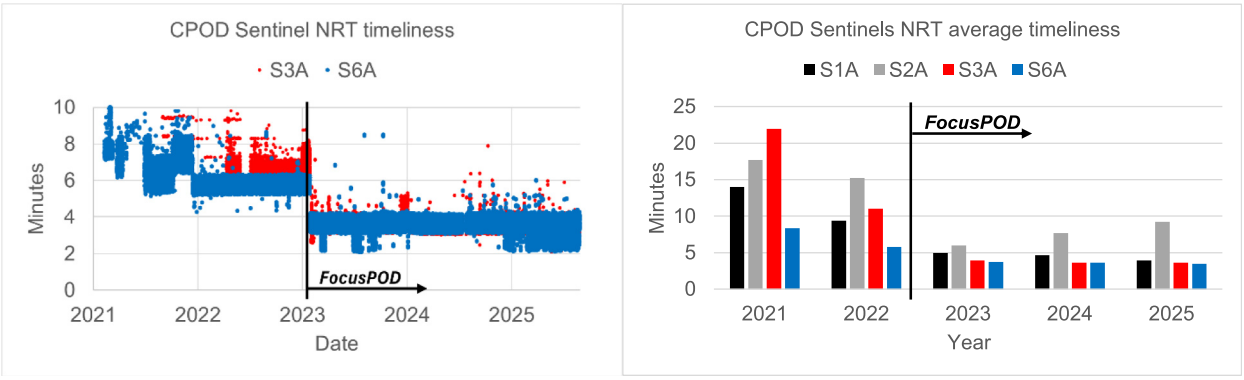


Fig. 7. CPOD timeliness of NRT products (minutes); left: S3A & S6A; right: yearly averages. After the introduction of FocusPOD, there is a drop of around 2 min.

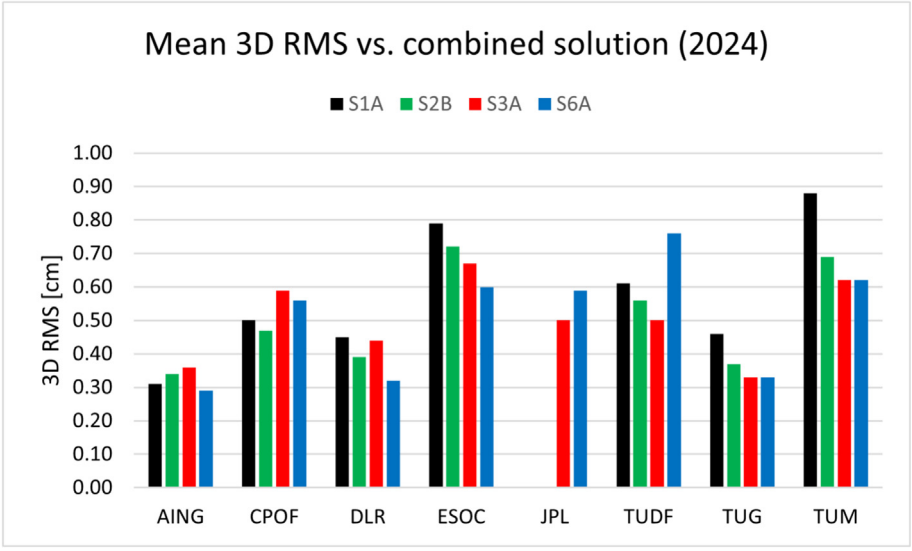


Fig. 8. Mean daily 3D RMS of orbit differences with respect to the combined solution for the year 2024. Bars correspond to individual satellites, one per mission: S1A, S2B, S3A, and S6A. The contributing analysis centres and POD software are: BSW (AING, TUM), *FocusPOD* (CPOF), GHOST (DLR), NAPEOS/EPNS (ESOC), Gipsy-X (JPL, TUDF), and GROOPS (TUG).

sistent orbit products using the most up-to-date models, algorithms, and input data. This work demonstrates that *FocusPOD* can efficiently support long-term reprocessing campaigns within a short time frame. The demonstration is conducted for the three altimetry missions currently processed by the CPOD Service (Sentinel-3A, Sentinel-3B, and Sentinel-6A) which are equipped with Laser Retro Reflectors (LRRs), enabling an independent validation of orbit accuracy. For Sentinel-3A&B, GPS-only observations were used, while for Sentinel-6A, Galileo-only observations were used prior to 16 May 2022 due to limitations in available GPS L2C bias products; from this date onward, combined GPS + Galileo observations were used.

The reprocessing campaign was carried out outside the operational CPOD environment, within a dedicated development environment. The computational setup consisted of an Intel(R) Xeon(R) E5-2640 v4 (2.40 GHz) processor with 20 cores and 32 GB of RAM, running Ubuntu 20.04.3. As the machine is shared with the development team, processing was scheduled to run automatically during nights and weekends, using 9 cores, each assigned to a specific satellite and day. An adaptive Python-based scheduling script monitored core availability and dynamically launched new tasks as soon as a core became free, ensuring optimal resource utilization.

All raw input data (GNSS observations, attitude quaternions, and manoeuvre information) and auxiliary GNSS products are stored on a dedicated 7.1 TB data volume. The files are indexed through a PostgreSQL database containing metadata that allows efficient querying and retrie-

Table 7
POD SW used by each Solution ID and their corresponding institution.

POD SW	Solution ID	Institution	GNSS products	Observations	Comment
BSW	AING	AIUB	CODE finals	GNSS	
	TUM	TU Munich	CODE finals	GNSS	
FocusPOD	CPOF	GMV	CODE finals	GNSS	
GHOST	DLR	DLR	CODE finals	GNSS	
GIPSY-X	JPL	JPL	JPL	GNSS	S3 and S6 only
	TUDF	TU Delft	JPL	GNSS	
GROOPS	TUG	TU Graz	TUG	GNSS	Raw processing
NAPEOS/EPNS	ESOC	ESA/ESOC	ESA	GNSS	

val of all files applicable to a given processing day. Upon completion of each orbit determination run, the processing metrics can be automatically written to a database for subsequent performance analysis.

In terms of computational performance, Sentinel-3 (GPS-only processing) required an average of 4 min per execution, while Sentinel-6A required 4 min 20 s (GPS + GAL). The campaign processed 3479, 2681, and 1741 daily arcs for Sentinel-3A, Sentinel-3B, and Sentinel-6A, respectively, covering the full mission lifetimes up to 31 August 2025. The total equivalent time to complete the reprocessing of all three missions was less than three days. Compared to the legacy CPOD software, the same workload required more than twice as much time, highlighting a significant improvement in processing efficiency.

Table 8 summarizes the orbit determination inputs and parameterization used for this reprocessing. Relative to the latest operational CPOD configuration (Table 6), the main differences are as follows:

- Two geopotential models from COST-G were used. The first model (G1) (Meyer, 2022), was employed for the period before the GRACE-FO launch (May 2018).
- The FES2022 model replaced FES2014 for ocean tides and loading.
- Atmospheric gravity was updated from AOD1B RL6 to AOD1B RL7.
- Solar activity indices were sourced from GFZ (Matzka et al., 2021) instead of the Report of Solar and Geophysical Activity (RSGA) from the National Oceanic and Atmospheric Administration (NOAA).

Table 8
POD parametrization of the reprocessing.

Name	Value
Geopotential	< 2019/01/01: G1; GSM-2_MODEL_GGFO_COSTG_BF01_01J2.gfc (Meyer, 2022) > 2019/01/01: G2; GSM-2_MODEL_GRFO_COSTG_BF01_02op_2506.gfc (Peter et al., 2020)
Ocean Tides	OTIDES.FES2022_120.amp005.atmAODa (CNES, 2024)
Ocean Loading	FES2022b_CMC.blq
Seasonal GM	ITRF2020-geocenter-motion.snx (Altamimi et al., 2023)
EOPs	fixed to IERS final products
GNSS inputs	2016/02/23–2019/12/31: CODE_REPRO_I20 (IGb14); GPS Only (Villiger et al., 2020) 2020/01/01–2022/11/26: CODE finals (IGb14); GPS Only (Dach et al., 2024) 2020/01/01–2022/11/26: CODE rapid (IGb14); GPS + GAL; only for S6A (Dach et al., 2024b) 2022/11/27–2025/08/31: CODE finals (IGS20); GPS + GAL (Dach et al., 2024)
IGS ANTEX	igs14_2247.atx (IGb14) (Rebischung and Schmid, 2016) igs20_2361.atx (IGS20) with GPS III extension (Conrad et al., 2023)
SEN ANTEX	sen14_2384.atx (IGb14) sen20_2384.atx (IGS20)
Atmospheric Gravity	AOD1B RL07 (Shihora et al., 2022)
Solar activity	GFZ (Matzka et al., 2021)
Attitude	Measured attitude quaternions were used, with gaps filled using simulated attitude. For S6A data prior to 8 July 2024, a yaw bias correction of −0.43° was applied to the quaternions, following Calliess et al. (2024).
Number of drag parameters	1 Estimated (S3) constrained with 0.3 1 Fixed (S6)
Solar radiation pressure	S3A 1 fixed to 0.97; S3B 1 fixed to 0.96; Values obtained from a long-term evaluation of estimated SRP coefficients. (Peter et al., 2020) S6A: 1 estimated
Empirical accelerations	S3_: 16 estimated per 32 h; Constant/Sine/Cosine in Along-track/Cross-track S6A: 8 estimated per 32 h; Constant/Sine/Cosine in Along-track/Cross-track
Phase ambiguity	Integer
Arc length	32 h

Regarding GNSS inputs, all products were obtained from CODE. At present, no fully consistent GNSS products series aligned to a single reference frame is available for the entire reprocessed interval considered here. Therefore, four distinct GNSS product series were used, ensuring internal consistency within each reprocessing interval and following the official transitions in reference frames and antenna models. Specifically, CODE_REPRO_I20 products (Villiger et al., 2020), aligned to IGb14, were used until the end of 2019, followed by CODE finals products (Dach et al., 2024) also aligned to IGb14 until 27 November 2022, and to IGS20 afterwards. For Sentinel-6A, which is processed using combined GPS and Galileo observations, the CODE rapid products (Dach et al., 2024b) were used until the switch to IGS20.

The use of CODE repro3 products was considered for this reprocessing; however, they do not provide code bias products for the GPS L2C signal, which are required for Sentinel-6A GPS processing. In addition, CODE repro3 is not fully aligned with either the IGb14 or IGS20 reference frame realizations and therefore does not resolve the fundamental issue of having GNSS products consistently aligned to a single reference frame over the entire reprocessing period. However, this comment does not preclude a more in-depth future analysis of repro3 for POD.

For the satellite attitude, the onboard attitude quaternions were employed for all missions; however, for Sentinel-6A data prior to 8 July 2024, a yaw bias of –0.43° was applied to correct the misalignment caused by the onboard Attitude and Orbit Control System (AOCS) (Calliess et al., 2024).

Table 9 summarizes the three main configurations used combining geopotential and GNSS products.

4.2. PCO/PCV estimation

Before reprocessing the Sentinel-3 & 6A missions, the Phase Centre Offset (PCO) and Phase Centre Variation (PCV) were estimated to align with the models and inputs described in the previous section. This was done with a tool provided by *FocusPOD*.

Two sets of PCO/PCV were produced, consistent with the IGb14 and IGS20 realizations. IGb14 was used until 26 November 2022, and IGS20 thereafter. For each realization, three months of S3A&B data and six months of S6A were processed (Table 10). The main selection criterion was to minimise solar-activity impacts; hence two IGb14 windows (S6A was launched in November 2020) and one IGS20 window. For S6A, an extended period was used to account for the observed correlation of the empirical signals with the beta angle (see Section 4.3).

The PCO/PCV estimation followed a two-step procedure. First, PCOs were estimated assuming zero PCV. Second, PCVs were estimated with the PCO fixed to the values from step one. Estimation uses all carrier-phase residuals from POD (Jäggi et al., 2009).

During PCO estimation, the X and Y components (aligned to the along-track and cross-track directions respectively) did not converge within five iterations; each iteration added nearly the same increment, likely due to correlations with residual force-model errors. In contrast, the Z component (aligned to the radial direction) converged completely within four to five iterations. Consequently, only the Z component was estimated, while X and Y were kept fixed to zero. Table 11 shows the converged Z components for each realization. From IGb14 to IGS20, the Sentinel-3A & 3B GPS L1/L2 solutions exhibit an increase of 3.15 & 3.14 mm in PCO-Z, whereas S6A shows an increase of 2.12 mm for GPS and a decrease of 6.19 mm for Galileo.

With PCO-Z fixed, PCVs were then estimated from the residuals of a final POD run. Iterating the PCV map improved convergence near zenith, but values near the local horizon (i.e., close to the X–Y plane) failed to stabilise and accumulated similar biases across iterations. Therefore, only the first PCV map was retained without further iteration. The resulting maps are shown in Fig. 9. Antennas on the same Sentinel platform exhibit similar PCV patterns, suggesting that the maps primarily capture spacecraft-specific environmental effects rather than intrinsic antenna differences (the antennas share manufacturer and type).

Table 9
Summary of three main configurations.

Case ID	G1_IGb14	G2_IGb14	G2_IGS20
Interval	2016/02/23 2018/12/31	2019/01/01 2022/11/26	2022/11/27 2025/08/31
Satellites	S3A, S3B	S3A, S3B S6A	S3A, S3B S6A
Geopotential	G1	G2	G2
GNSS	CODE REPRO	S3: CODE finals S6: CODE rapid	CODE finals
Frame	IGb14	IGb14	IGS20
Attitude	S3: Quaternions	S3: Quaternions S6A: Quaternions + yaw bias	S3: Quaternions S6A: Quaternions (+ yaw bias)

Table 10
Time interval used to estimate PCO/PCV.

Period	Frame	Satellites (main antenna)	Frequencies
2020-05-31 – 2020-09-05	IGb14	S3A, S3B	L1, L2
2022-05-29 – 2022-11-25	IGb14	S6A	L1, L2, E1, E5a
2023-05-28 – 2023-11-24	IGS20	S3A, S3B	S3: L1, L2
		S6A	S6: L1, L2, E1, E5a

Table 11
PCO – Z component for each nominal antenna onboard Sentinels-3 and 6.

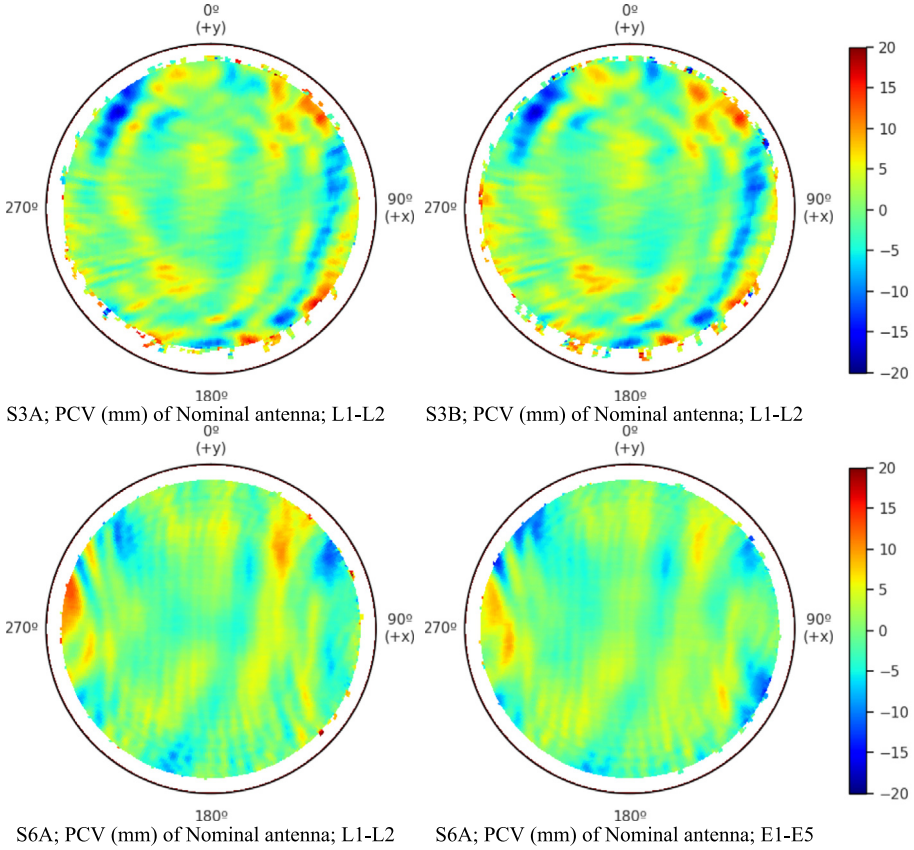
Satellite/GNSS	IGb14 [mm]	IGS20 [mm]	Differences [mm]
S3A (GPS)	75.96	79.11	3.15
S3B (GPS)	73.51	76.65	3.14
S6A (GPS)	69.57	71.69	2.12
S6A (GALILEO)	89.08	82.89	–6.19

4.3. Results

To assess the reprocessed orbits, they are compared against the CPOD combined orbit solution. The combined solution is constructed as a weighted mean of several independent orbit solutions generated by different analysis centres using distinct POD software and input configurations. The combination procedure is initialized using an unweighted mean solution, after which each individual solution is compared to the combined orbit to derive centre- and day-specific weights based on their level of agreement. This process can be iterated to achieve a stable combined solution. The weights are computed per analysis centre and per day, rather than per individual state vector. The solutions involved in the combined solutions can be found in Table 7.

Figs. 10 and 11 show the daily 3D and radial RMS differences between the reprocessed orbits of Sentinel-3A and Sentinel-3B (left panels) and Sentinel-6A (right panels) with respect to the combined solution. Note that comparisons with the combined orbit solution are used to assess consistency with other analysis centres. Orbit accuracy is later evaluated independently using SLR residuals, which are particularly sensitive to radial orbit errors.

Two noteworthy features are apparent in these plots. First, the differences for Sentinel-3 gradually improve from 2016 to 2020, after which they become more stable. This trend is not due to the reprocessing itself but to the evolu-



S3A; PCV (mm) of Nominal antenna; L1-L2 S3B; PCV (mm) of Nominal antenna; L1-L2

S6A; PCV (mm) of Nominal antenna; L1-L2 S6A; PCV (mm) of Nominal antenna; E1-E5

Fig. 9. PCV maps for each nominal antenna onboard Sentinels-3A, -3B and -6A.

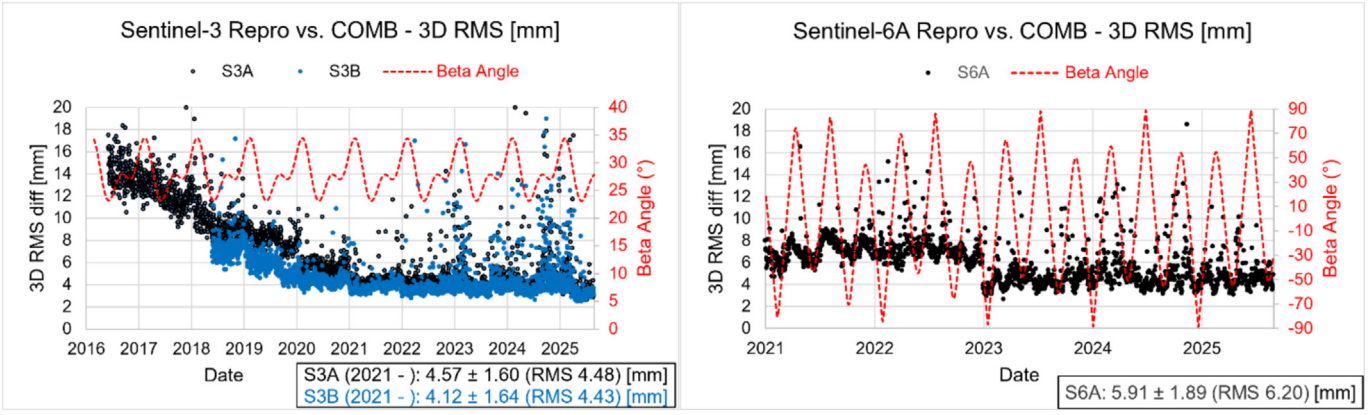


Fig. 10. Sentinel-3A, 3B & 6A 3D RMS differences of reprocessed orbit vs. the operational combined solution (COMB). Left panel shows Sentinel-3A & B, while right panel shows Sentinel-6A. Red dotted curve is the beta angle. Improving statistics in Sentinel-3 from 2016 to 2020 are due to the changing quality of the combined solution used. Legend provides the mean, standard deviation and RMS of each corresponding satellite.

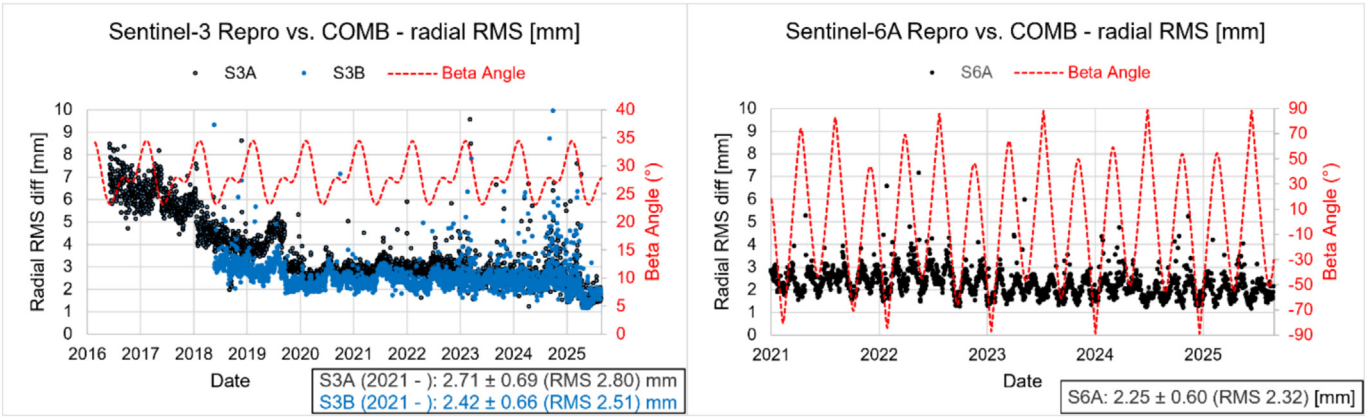


Fig. 11. Sentinel-3A, 3B & 6A radial RMS differences of reprocessed orbit vs. the operational combined solution (COMB). Left pannel shows Sentinel-3A & B, while right panel shows Sentinel-6A. Red dotted curve is the beta angle. Improving statistics in Sentinel-3 from 2016 to 2020 are due to the changing quality of the combined solution used. Legend provides the mean, standard deviation and RMS of each corresponding satellite. Sentinel-6A values exhibit a periodic behaviour related with the beta angle, probably due to SRP modelling deficiencies.

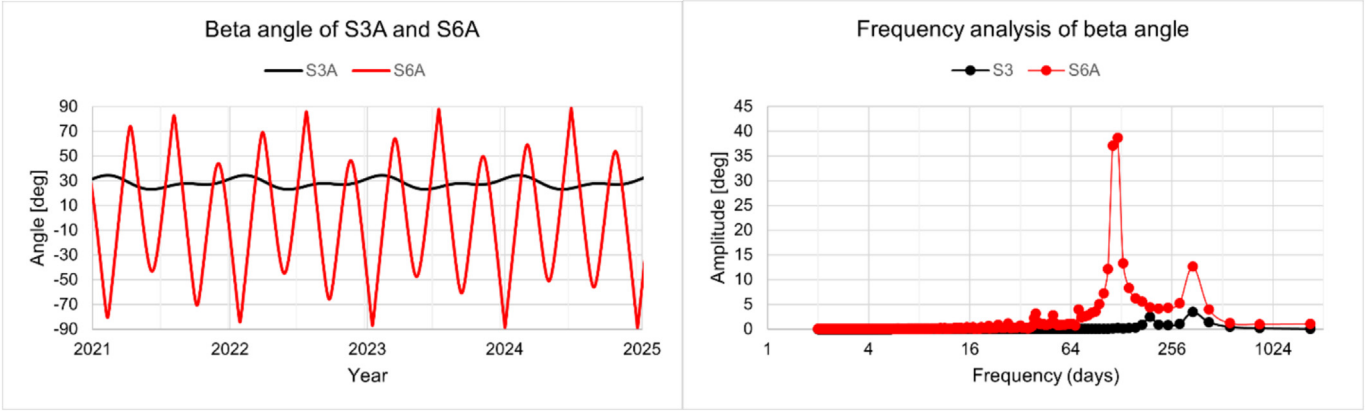


Fig. 12. Beta-angle (left) and its dominant frequencies (right) of Sentinel-3A and Sentinel-6A. Sentinel-6A shows a noticeable dominant frequency around 117.8 days.

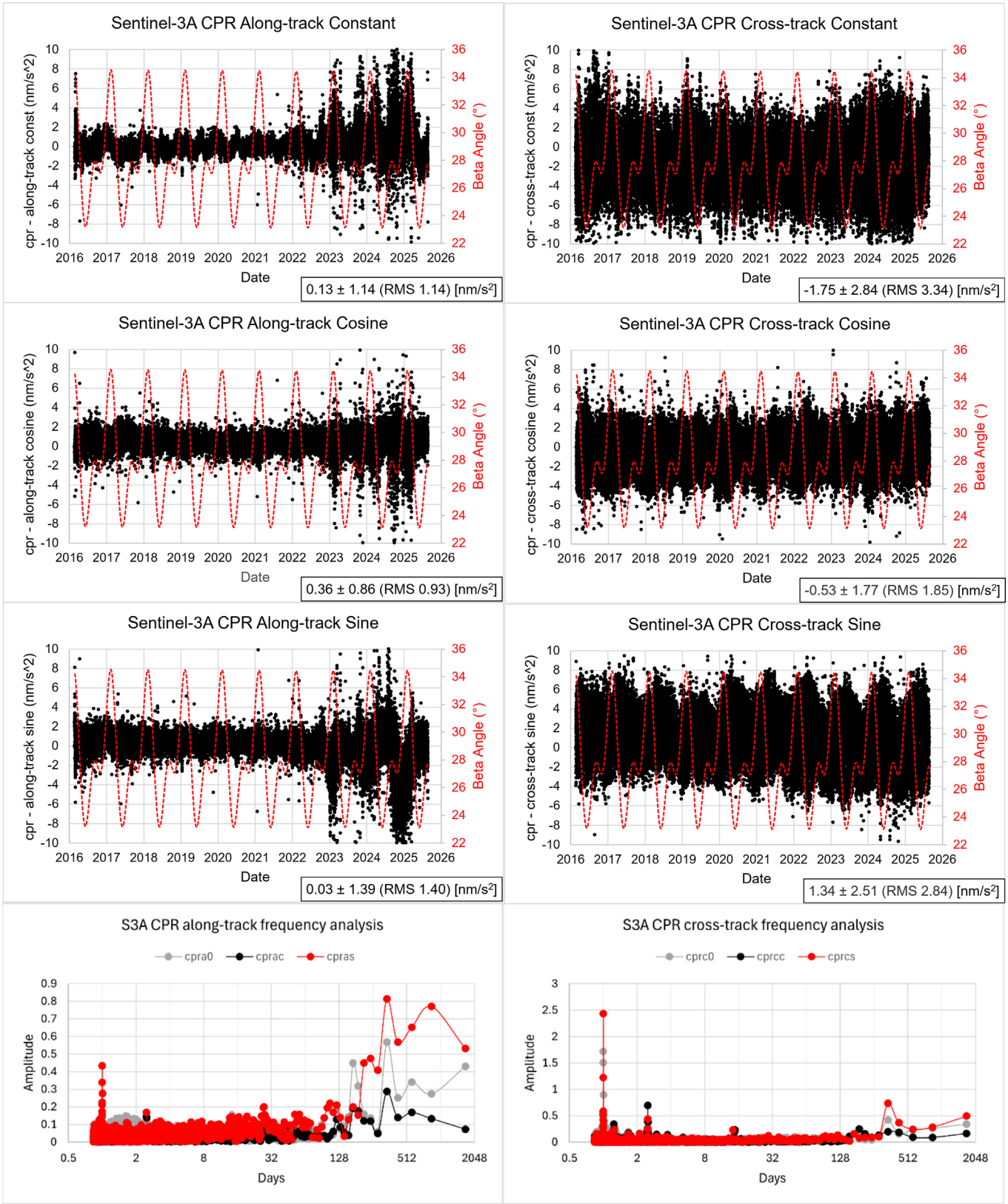


Fig. 13. Estimated CPR of Sentinel-3A for the along-track (left) and cross-track (right) components, including the constant, cosine, and sine terms. The lower panels show the dominant frequencies identified for each component, with a dominant frequency around 1 day in the cross-track constant and sine components.

tion of the combined orbit product over that period. The CPOD Quality Working Group (QWG) (Fernández et al., 2024) progressively updated its orbit standards and input products, leading to the observed improvement in consistency. The absence of a reprocessed combined orbit solution for the earlier years explains the apparent temporal dependence.

Second, a noticeable increase in the noise level (particularly in the 3D RMS) is observed during 2023 and 2024. This can be attributed to the heightened solar activity during this period, which led to a higher frequency of orbit manoeuvres (for example, Sentinel-3A performed five manoeuvres in 2020 and twenty-eight in 2024).

The global statistics confirm the high internal consistency of the results: for the period January 2021 to August 2025, Sentinel-3A and Sentinel-3B exhibit mean 3D RMS values of 4.57 mm and 4.12 mm, respectively, and radial RMS values of 2.71 mm and 2.42 mm. For Sentinel-6A, the corresponding values are 5.91 mm (3D) and 2.25 mm (radial).

An additional and intriguing feature is the sinusoidal pattern visible in the Sentinel-6A results, particularly in the radial component, with a period of approximately 58.8 days. This signal is related to draconitic periodicities, i.e. variations associated with the repeat cycle of the satellite's orbital plane with respect to the Sun (β-angle). For Sentinel-6A, the β-angle exhibits a dominant period of approximately 117.8 days, corresponding to the fundamental Sun–orbit geometry cycle, while the 58.8-day signal represents its first harmonic. This period is also close to one-sixth of the GPS draconitic period, a known feature in GPS orbit products related to the six orbital planes. At present, this behaviour is most likely related to deficiencies in the solar radiation pressure (SRP) modelling, although a detailed characterization of the underlying mechanisms is still under investigation (see Lemoine et al., 2026).

To complement the previous discussion, Fig. 12 shows the evolution of the β-angle on the left (i.e., the elevation of the Sun above the orbital plane) and the corresponding frequency analysis on the right, for Sentinel-3A (which is in a Sun-synchronous polar orbit) and Sentinel-6A. Sentinel-6A exhibits large β-angle variations, with a dominant frequency of approximately 117.8 days and a correspondingly large amplitude. In contrast, Sentinel-3A shows much smaller β-angle variations.

Fig. 13 shows the estimated CPR of Sentinel-3A in the along-track (left) and cross-track (right) directions, including the constant, cosine, and sine terms (from top to bottom). The bottom panels present the corresponding frequency analysis of these empirical accelerations.

Several noteworthy features can be identified. First, a marked increase in the dispersion of the along-track CPRs is observed from 2023 onwards, which can be attributed to the increased solar activity during this period. Second, the harmonic analysis reveals distinct dominant frequencies, particularly in the cross-track empirical accelerations. In particular, a pronounced peak at high frequencies is visible in the constant and sine components.

Fig. 14 shows a one-month zoom in 2019 for Sentinel-3 CPRs cross-track constant and sine empirical components. With the values connected by a line, it becomes evident that these empirical terms are absorbing unmodelled effects, with a dominant period of one day. Preliminary analyses within the CPOD QWG suggest that this pattern is related to geocentre motion. Although its subdaily variations probably cannot be estimated from LEO observations, future work will focus on implementing the capability to estimate at least daily geocentre coordinates within *FocusPOD*.

Fig. 15 presents the evolution of the estimated CPRs for Sentinel-6A, which display a pronounced sinusoidal pattern in most components. A harmonic analysis reveals dominant frequencies of approximately 58.8 days for the cross-track cosine, and cross-track sine terms; 117.8 days for the along-track cosine and sine terms.

This behaviour suggests a deficiency in the current solar radiation pressure (SRP) modelling. Sentinel-6A features a configuration in which the solar panels are rigidly fixed to

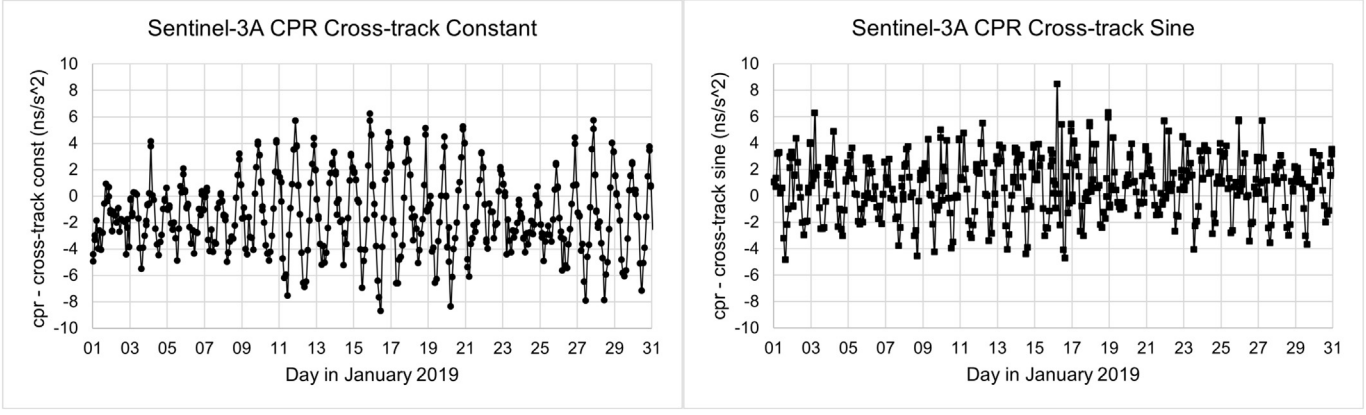


Fig. 14. Estimated CPRs of Sentinel-3A in the cross-track direction, for the constant (left) and sine (right) components. It shows a signal with a frequency of 1 day, which is attributed to the geocentre motion not being estimated.

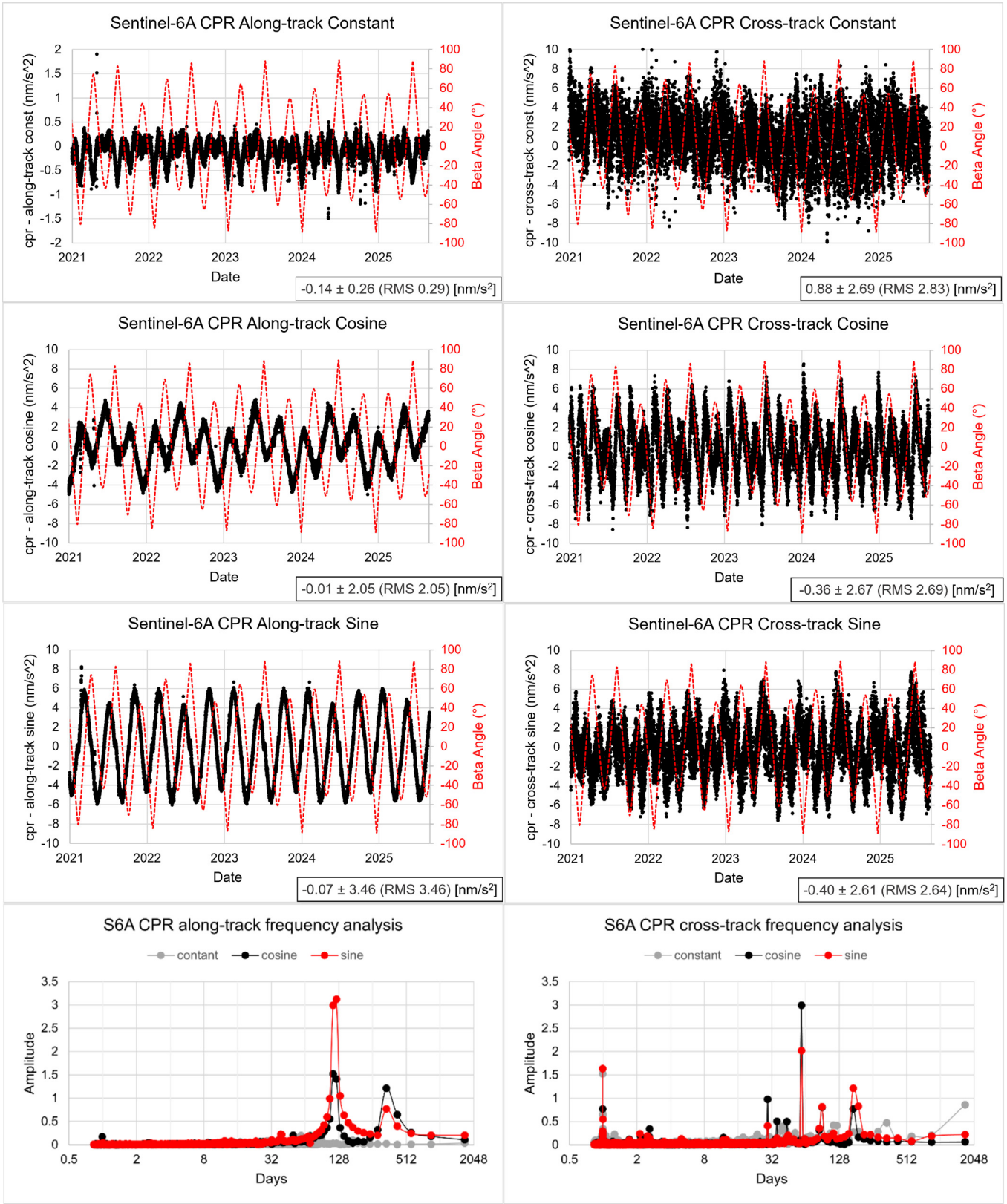


Fig. 15. Estimated CPR of Sentinel-6A for the along-track (left) and cross-track (right) components, including the constant, cosine, and sine terms. The lower panels show the dominant frequencies identified for each component, with dominant frequencies around 117.8 days for the along-track cosine and sine terms and 58.8 days for the cross-track cosine and sine terms.

Fig. 16. Sentinel-6 (left) (credit: ESA/ATG medialab) and Sentienl-3 (right) (credit: ESA-Pierre Carril). The roof of Sentinel-6A together with the large β-angle variations caused periodic shadowing of the nadir part of the satellite body. Sentinel-3 instead, does not suffer these periodic shadowing.

Table 12
SLR network used with estimated biases. Note that WETL has two values, one of each frequency used.

Monument	Code	Location	Frequency [nm]	Average Bias [mm]	Standard deviation [mm]
7090	YARL	Yarragadee, Australia	532	5.41	1.88
7105	GODL	Greenbelt, Maryland	532	−7.56	4.18
7110	MONL	Monument Peak, California	532	−3.82	2.65
7119	HA4T	Haleakala, Hawaii	532	8.64	3.39
7501	HARL	Hartebeesthoek, South Africa	532	−4.05	4.02
7810	ZIML	Zimmerwald, Switzerland	532	5.17	5.04
7825	STL3	Mt Stromlo, Australia	532	6.86	1.75
7839	GRZL	Graz, Austria	532	2.11	1.78
7840	HERL	Herstmonceux, United Kingdom	532	−4.85	1.84
7841	POT3	Potsdam, Germany	532	−2.36	5.34
8834	WETL	Wettzell, Germany	532	−1.61	2.31
8834	WETL	Wettzell, Germany	1064	−1.74	2.55

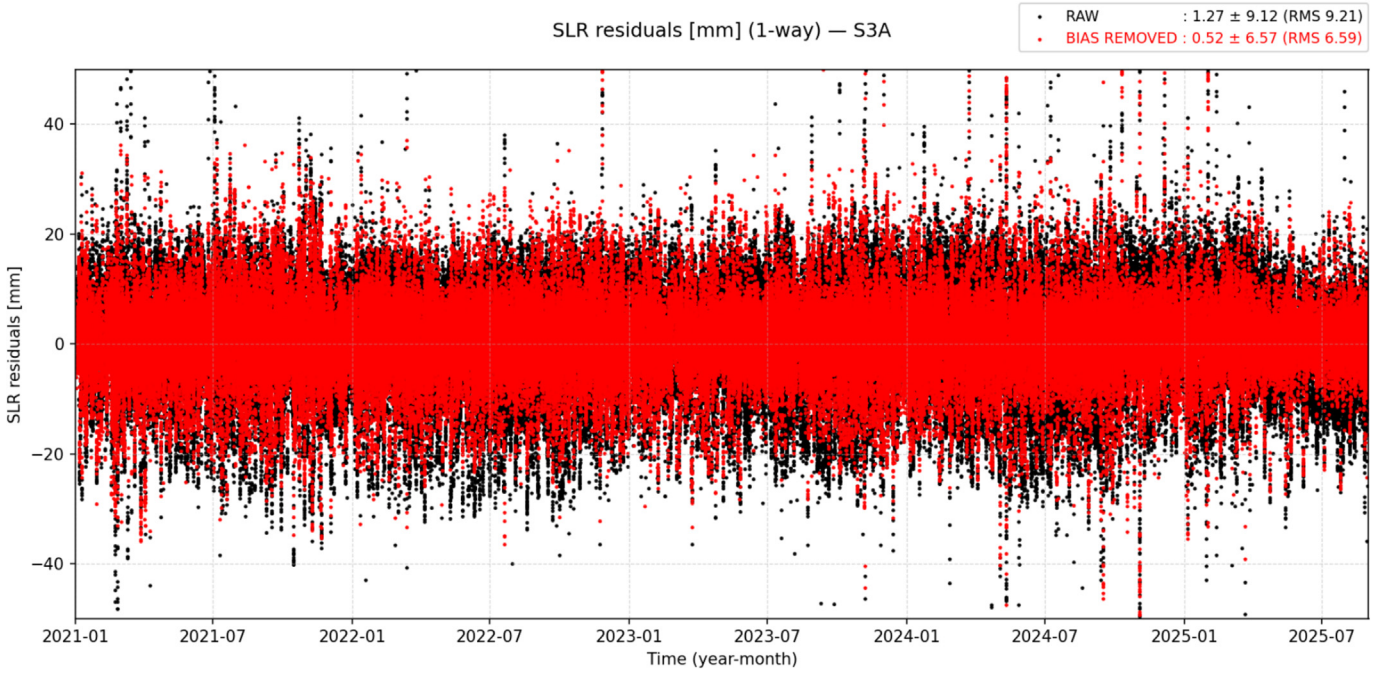


Fig. 17. Sentinel-3A SLR residuals (1-way [mm]); raw residuals and after removing a station bias.

the satellite body (Fig. 16 left), partially shadowing elements that point toward nadir when the β-angle is high. As described above, the β-angle exhibits a dominant period of 117.8 days, the latter coinciding with the strongest period observed in the CPRs, while the 58.8 days is its first harmonic. The macro-model used in the analysis (from Montenbruck et al., 2021) does not fully capture this variability, causing the CPR estimates to absorb these unmodelled effects. Although this has no significant impact on the overall orbit differences (except the clear signal on the radial component), it clearly highlights an area where the physical modelling could be improved.

The comparisons with the combined orbit solution primarily assess the consistency of the reprocessed orbits with those from other analysis centres, whereas the analysis of SLR residual analysis provides an independent assessment of orbit accuracy, particularly sensitive to errors in the radial component.

The SLR residual analysis was performed using data from 11 ILRS stations (see Table 12). Prior to computing the residuals, station- and month-dependent range biases were estimated using SLR observations to Sentinel-3A, Sentinel-3B, and Sentinel-6A, in order to minimize correlations with orbit errors. The bias estimation used the SLRF2020 (ILRS, 2023) frame and the reprocessed orbits. The resulting average biases and their corresponding standard deviations are summarized in Table 12. For most stations, the estimated biases remain within ±5 mm.

Once estimated, the SLR residuals are recomputed after applying the bias correction. Fig. 17 illustrates the one-way SLR residuals for Sentinel-3A before and after removing the estimated station biases, yielding RMS values of 9.21 mm and 6.59 mm, respectively, for the period 2021–2025. This analysis includes all available SLR observations. However, since SLR is primarily sensitive to radial orbit errors, the results should be interpreted mainly in terms of the radial component. It is therefore customary to examine the dependence of the residuals on the elevation cut-off angle, as discussed below.

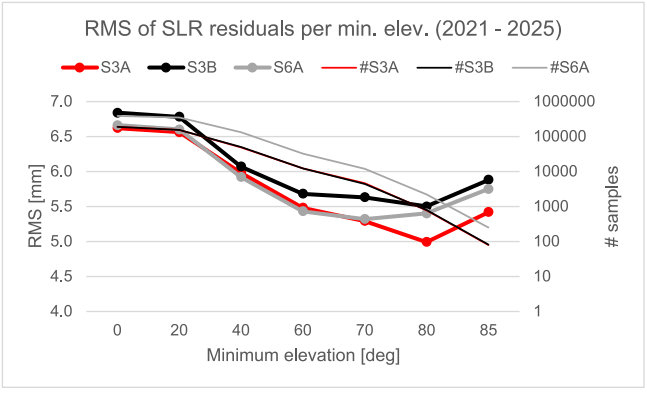


Fig. 18. Elevation dependent RMS of one-way SLR residuals for S3A, S3B and S6A over the period 2021–2025. The number of samples contributing to each elevation bin is shown on the left y-axis using a logarithmic scale.

Fig. 18 depicts the elevation-dependent behaviour of the one-way SLR residual RMS, providing insight into the directional sensitivity of the SLR observations to orbit errors. When only SLR observations above 80° elevation are considered, the RMS of the residuals lies in the range of 5.0–5.5 mm, which is mainly representative of the radial orbit error component. Including lower-elevation observations increases the RMS to approximately 6.5–7.0 mm, since the SLR line of sight becomes more sensitive not only to radial errors but also to along-track and cross-track orbit errors. For elevation angles above 85°, the RMS increases by about 0.5 mm; however, the number of observations in this regime drops sharply to around 100 samples accumulated over nearly five years, which limits the statistical robustness of these estimates. For this reason, the number of SLR observations contributing to each elevation bin is shown to provide context for the RMS values.

It is also noteworthy that the SLR residual analysis yields larger RMS values than those obtained from comparisons with the combined orbit solution. For example, for Sentinel-6A, the radial RMS with respect to the combined solution is approximately 2.4 mm, whereas the RMS derived from SLR residuals is about 5.5 mm. Similar discrepancies can be found for Sentinel-3A & B. This factor-of-two difference highlights the distinction between orbital consistency (or precision) and absolute orbit accuracy. The relatively small discrepancies with respect to the combined solution partly reflect the use of similar dynamical models, background models, and GNSS products across the contributing analysis centres, which reduces the independence of the solutions. As a result, inter-centre comparisons tend to underestimate the true orbit error. In contrast, SLR observations provide an independent measurement technique, particularly sensitive to the radial component, and therefore offer a more realistic assessment of the absolute orbit accuracy.

5. Conclusions and way forward

This article has presented *FocusPOD*, a new POD and geodesy library designed and developed at GMV. Built in modern C++ and interoperable with Python, *FocusPOD* combines state-of-the-art software engineering with the methodological rigor required for scientific and operational geodesy. The library has been successfully integrated into several infrastructures, particularly the CPOD Service.

The results obtained in operational and reprocessing contexts confirm that *FocusPOD* delivers orbit solutions with centimetre- to millimetre-level precision, meeting or exceeding the requirements of the Copernicus missions. Its modular data model and unified architecture allow all processing stages to be executed in a single workflow, eliminating intermediate I/O operations and preserving floating-point precision throughout computation. The reprocessing of Sentinel-3A, -3B, and -6A missions demonstrated both the efficiency and scientific accuracy of the system, with mean orbit differences of ~5 mm (3D RMS) and

~2–3 mm (radial RMS) with respect to combined solutions over multi-year intervals. Independent validation using Satellite Laser Ranging (SLR) confirms orbit accuracy at the 5–5.5 mm level in the radial component.

The analysis of Sentinel-3A identified a pronounced one-day periodic signal in the cross-track constant and sine CPR components, most likely associated with unmodelled geocentre motion, which is not currently estimated. For Sentinel-6A, subtle periodic signatures were detected in the CPRs, indicating residual deficiencies in the solar radiation pressure modelling.

The modular architecture of *FocusPOD* provides a foundation for future extensions toward multi-station GNSS processing, multi-technique combinations, and reference system realization. While most of these capabilities are already supported at the architectural level, their full scientific validation remains ongoing. Future work will focus on consolidating these developments within operational and geodetic contexts.

In summary, *FocusPOD* represents a unified and operationally validated framework for GNSS-based LEO POD. The results presented in this study demonstrate its robustness, scalability, and numerical performance in operational and large-scale reprocessing contexts. While the underlying architecture is designed to support broader geodetic parameter estimation and multi-technique processing, these capabilities are currently under validation and will be addressed in future work.

Data availability

Sentinels GNSS observations and auxiliary data for POD, the operational orbits generated by the CPOD Service as well as the combined solutions are openly shared at the Copernicus Data Space Ecosystem (https://dataspace.copernicus.eu). Precise GNSS orbit, clock and biases as used in the study are publicly available from CDDIS (https://cddis.nasa.gov/archive/gnss/data/) and CODE (https://aiub.unibe.ch).

Declaration of competing interest

The authors declare that they have no conflict of interest.

Acknowledgements

The Copernicus POD Service is financed under ESA contract No. 4000139509/22/I-BG, which is gratefully acknowledged. The work performed in the frame of this contract is carried out with funding by the European Union. The views expressed herein can in no way be taken to reflect the official opinion of either the European Union or the European Space Agency.

The authors gratefully acknowledge the provision of Sentinel GNSS observations by the Copernicus Earth observation component of the European Union's (EU) Space program, as well as the collection and sharing of SLR measurements through the International Laser Ranging Service (ILRS) (Pearlman et al., 2019). Auxiliary products for the Sentinel data analysis have been provided by the International GNSS Service (IGS) (Johnston et al., 2017) and the Center for Orbit Determination in Europe (CODE) (Dach et al., 2024a). Combined solutions of Sentinels have been generated by the Copernicus POD (CPOD) Service (Fernández et al., 2024) using independent orbits generated by its Quality Working Group (QWG). The COST-G geopotential has been provided by the International Gravity Field Service (IGFS) (Barzaghi, 2017) and U. Meyer from AIUB. The support of all institutions and individuals is likewise acknowledged.

References

Altamimi, Z., Rebischung, P., Collilieux, X., Métivier, L., Chanard, K., 2023. ITRF2020: an augmented reference frame refining themodeling of nonlinear station motions. J. Geod. 97, 47. https://doi.org/10.1007/s00190-023-01738-w.

Arnold, D., Meindl, M., Beutler, G., Dach, R., Lutz, S., Prange, L., Sośnica, K., Mervart, L., Jäggi, A., 2015. CODE's new solar radiation pressure model for GNSS orbit determination. J. Geod. 89, 775–791. https://doi.org/10.1007/s00190-015-0814-4.

Bancroft, S., 1985. An algebraic solution of the GPS equations. IEEE Trans. Aerosp. Electron. Syst. AES-21 (1), 56–59. https://doi.org/10.1109/TAES.1985.310538.

Barzaghi, R., 2017. International Gravity Field Service (IGFS). In: Sideris, M.G. (Ed.), Encyclopedia of Geodesy. Encyclopedia of Earth Sciences Series. Springer, Cham. https://doi.org/10.1007/978-3-319-02370-0_18-1.

Bertiger, W., Bar-Sever, Y., Dorsey, A., Haines, B., Harvey, N., Hemberger, D., Heflin, M., Lu, W., Miller, M., Moore, A.W., Murphy, D., Ries, P., Romans, L., Sibois, A., Sibthorpe, A., Szilagyi, B., Vallisneri, M., Willis, P., 2020. GipsyX/RTGx, a new tool set for space geodetic operations and research. Adv. Space Res. 66 (3), 469–489. https://doi.org/10.1016/j.asr.2020.04.015.

Boomkamp, H., 2006. Software modernization in support of LEO and Multi-Constellation processing. Poster at IGS Workshop 2006. https://files.igs.org/pub/resource/pubs/06_darmstadt/IGS%20Posters%20PDF/p_3_Boomkamp.pdf. (Accessed 27 March 2026).

Calliess, D., Montenbruck, O., Wermuth, M., Reichinger, H., 2024. Long-term analysis of Sentinel-6A orbit determination: insights from three years of flight data. Adv. Space Res. 74 (7), 3011–3027. https://doi.org/10.1016/j.asr.2024.06.043.

Carrou, J.P., 1986. Zoom Software: error analysis and accurate orbit restitution at CNES. In: Bhatnagar, K.B. (Ed.), Space Dynamics and Celestial Mechanics. Astrophysics and Space Science Library. Springer, Dordrecht. https://doi.org/10.1007/978-94-009-4732-0_36.

CNES, 2022. RINEX DORIS 3.0 (Issue 1.7). Technical Report CNES. URL: http://ftp.ids-doris.org/pub/ids/data/RINEX_DORIS.pdf.

CNES, 2024. FES2022 (Finite Element Solution) Ocean Tide (Version 2022). https://doi.org/10.24400/527896/A01-2024.004.

Conrad, A., Desai, S., Haines, B., et al., 2023. Extending the GPS IIIA antenna calibration for precise orbit determination of low Earth orbit satellites. J. Geod. 97 (4), 35. https://doi.org/10.1007/s00190-023-01718-0.

Dach, R., Lutz, S., Walser, P., Fridez, P. (Eds.), 2015. Bernese GNSS Software Version 5.2. User Manual. Astronomical Institute, University of Bern, Bern Open Publishing. https://doi.org/10.7892/boris.72297 (ISBN: 978-3-906813-05-9).

Dach, R., Schaer, S., Arnold, D., Brockmann, E., Kalarus, M.S., Lasser, M., Stebler, P., Jäggi, A., 2024. In: CODE Final Product Series for the

IGS. Astronomical Institute, University of Bern. https://doi.org/10.48350/197025.

Dach, R., Schaer, S., Arnold, D., Brockmann, E., Kalarus, M.S., Lasser, M., Stebler, P., Jäggi, A., 2024b. In: CODE Product Series for the IGS MGEX Project. Astronomical Institute, University of Bern. https://doi.org/10.48350/197028.

Dobslaw, H., Bergmann-Wolf, I., Dill, R., Poropat, L., Thomas, M., Dahle, C., Esselborn, S., König, R., Flechtner, F., 2017. A new high-resolution model of non-tidal atmosphere and ocean mass variability for de-aliasing of satellite gravity observations: AOD1B RL06. Geophys. J. Int. 211 (1), 263–269. https://doi.org/10.1093/gji/ggx302.

EC, 2016. European GNSS (Galileo) Open Service – Ionospheric correction algorithm for Galileo single frequency users, Issue 1.2, Sept. 2016, European Commission. https://www.gsc-europa.eu/sites/default/files/sites/all/files/Galileo_Ionospheric_Model.pdf.

Fernández, J., Peter, H., Fernández, C., Berzosa, J., Fernández, M., Bao, L., Muñoz, M.A., Lara, S., Terradillos, E., Féménias, P., Nogueira, C., 2024. The copernicus POD service. Adv. Space Res. 74 (6), 2615–2648. https://doi.org/10.1016/j.asr.2024.02.056.

Fernández, M., Peter, H., Arnold, D., Duan, B., Simons, W., Wermuth, M., Hackel, S., Fernández, J., Jäggi, A., Hugentobler, U., Visser, P., Féménias, P., 2022. Copernicus Sentinel-1 POD reprocessing campaign. Adv. Space Res. 70 (2), 249–267. https://doi.org/10.1016/j.asr.2022.04.036.

Folkner, W.M., Williams, J.G., Boggs, D.H., 2009. The Planetary and Lunar Ephemeris DE421. JPL Interplanet. Netw. Prog. Rep., 42–178

Gini, F., Hauschild, A., 2025. The Receiver Independent Exchange Format, version 4.02. International GNSS Service (IGS). RINEX Working Group and Radio Technical Commission for Maritime Services Special Committee 104 (RTCM-SC104).

Gipson, J., 2021. vgosDB Manual. IVS Working Group IV on Data Structures. https://ivscc.gsfc.nasa.gov/IVS_AC/vgosDB/vgosDB_format_2021Sep20.pdf.

GMV GSharp, 2023. GMV's Safe and High-Accuracy solution for multipurpose applications and services. https://www.gmv.com/sites/default/files/content/file/2023/12/12/111/au_br1_gmv-gsharp_en.pdf. (Accessed 27 March 2026).

Gurtner, W., Estey, L., 2007. RINEX – the receiver independent exchange format, version3.0. International GNSS Service (IGS). RINEX Working Group and Radio Technical Commission for Maritime Services Special Committee 104 (RTCM-SC104).

International Laser Ranging Service (ILRS), 2023. SLRF2020 – International Laser Ranging Service (ILRS) extension of the International Terrestrial Reference Frame 2020 (ITRF2020) TRF Model with additional SLR. NASA Crustal Dynamics Data Information System. https://doi.org/10.5067/SLR/SLRF2020_001 (Accessed 27 March 2026).

Hilla, S., 2010. The Extended Standard Product 3 Orbit Format (SP3-c). https://files.igs.org/pub/data/format/sp3c.txt.

Hilla, S., 2016. The Extended Standard Product 3 Orbit Format (SP3-d). https://files.igs.org/pub/data/format/sp3d.pdf.

IERS, 2006. SINEX - Solution (Software/technique) INdependent EXchange Format Version 2.02. https://www.iers.org/IERS/EN/Organization/AnalysisCoordinator/SinexFormat/sinex.html.

ILRS, 2018. Consolidated Laser Ranging Prediction Format (CPF) version 2.0. For the ILRS Data Formats and Procedures Working Group. https://ilrs.gsfc.nasa.gov/docs/2018/cpf_2.00h-1.pdf.

ILRS, 2019. Consolidated Laser Ranging Data Format (CRD) version 2.01. For the ILRS Data Formats and Procedures Working Group. https://ilrs.gsfc.nasa.gov/docs/2022/crd_v2.01e3.pdf.

Jäggi, A., Dach, R., Montenbruck, O., Hugentobler, U., Bock, H., Beutler, G., 2009. Phase center modeling for LEO GPS receiver antennas and its impact on precise orbit determination. J. Geod. 83 (12), 1145–1162. https://doi.org/10.1007/s00190-009-0333-2.

Johnston, G., Riddell, A., Hausler, G., 2017. The International GNSS Service. In: Teunissen, P.G., Montenbruck, O. (Eds.), Springer Handbook of Global Navigation Satellite Systems. Springer, pp. 967–982. https://doi.org/10.1007/978-3-319-42928-1_33 (Chapter 33).

Kawate, K., Igarashi, Y., Yamada, H., Akiyama, K., Okeya, M., Takiguchi, H., Murata, M., Sasaki, T., Matsushita, S., Miyoshi, S., Miyoshi, M., Kogure, S., 2023. MADOCA: Japanese precise orbit and clock determination tool for GNSS. Adv. Space Res. 71 (10), 3927–3950. https://doi.org/10.1016/j.asr.2023.01.060.

Knocke, P.C., Ries, J.C., Tapley, B.D., 1988. Earth radiation pressure effects on satellites. AIAA/AAS Astrodyn. Conf. 1988, 577–587. https://doi.org/10.2514/6.1988-4292.

Lemoine, F.G., Kenyon, S.C., Factor, J.K., Trimmer, R.G., Pavlis, N.K., Chinn, D.S., Cox, C.M., Klosko, S.M., Luthcke, S.B., Torrence, M. H., Wang, Y.M., Williamson, R.G., Pavlis, E.C., Rapp, R.H., Olson, T.R., 1998. The Development of the Joint NASA GSFC and the National IMagery and Mapping Agency (NIMA) Geopotential Model EGM96; GSFC Goddard Space Flight Center; Greenbelt, Md. https://ntrs.nasa.gov/citations/19980218814

Lemoine, F.G., Zelensky, N.P., Beckley, B.D., Mitchum, G.T., Yang, X., Nicholas, J.B., 2026. Precise orbit determination for TOPEX/Poseidon, Jasons 1,2,3 and Sentinel-6A and the std2400 series of orbits. Adv. Space Res. https://doi.org/10.1016/j.asr.2025.12.112.

Lemoine, J.M., Biancale, R., Reinquin, F., Bourgogne, S., Gégout, P., 2019. CNES/GRGS RL04 Earth gravity field models, from GRACE and SLR data. GFZ Data Serv. https://doi.org/10.5880/ICGEM.2019.010.

Lemoine, J.-M., Bourgogne, S., Capdeville, Reinquin, H.F., 2023. New mean gravity field model CNES_GRGS.RL05MF_combined_GRACE_SLR_DORIS. IDS AWG, 2023/04/18. IDSAWG202304-LemoineJM-NewMeanGravityModel.pdf.

Li, X., Huang, J., Li, X., Yuan, Y., Zhang, K., Zheng, H., Zhang, W., 2024. GREAT: a scientific software platform for satellite geodesy and multi-source fusion navigation. Adv. Space Res. 74 (4), 1751–1769. https://doi.org/10.1016/j.asr.2024.05.044.

Loyer, S., Montenbruck, O., Hilla, S., 2019. ORBEX: The Orbit Exchange Format, draft version 0.09, Available online: http://acc.igs.org/misc/ORBEX009.pdf.

Lyard, F.H., Allain, D.J., Cancet, M., Carrère, L., Picot, N., 2021. FES2014 global ocean tides atlas: design and performance. Ocean Sci. 17, 615–649. https://doi.org/10.5194/os-17-615-2021.

Mackenzie, R., 2021. GODOT Flight Dynamics Infrastructure Software for operations and analysis. In: 1st European Workshop on Space Flight Dynamics Services, Systems and Operations, ESA-ESOC, Darmstadt, Germany, September 2021.

Marty, J.C., Loyer, S., Perosanz, F., Mercier, F., Bracher, G., Legresy, B., Portier, L., Capdeville, H., Fund, F., Lemoine, J.M., Biancale, R., 2011. GINS: the CNES/GRGS GNSS scientific software. https://ids-doris.org/images/documents/report/publications/GINS_GRGSsoftware-Marty-2011.pdf. (Accessed 27 March 2026).

Matzka, J., Stolle, C., Yamazaki, Y., Bronkalla, O., Morschhauser, A., 2021. The geomagnetic Kp index and derived indices of geomagnetic activity. Space Weather 19 (5) https://kp.gfz-potsdam.de/app/files/Kp_ap_Ap_SN_F107_since_1932.txt. https://doi.org/10.1029/2020SW002641.

Mayer-Gürr, T., Behzadpour, S., Eicker, A., Ellmer, M., Koch, B., Krauss, S., Pock, C., Rieser, D., Strasser, S., Süsser-Rechberger, B., Zehentner, N., Kvas, A., 2021. GROOPS: a software toolkit for gravity field recovery and GNSS processing. Comput. Geosci. 155, 104864. https://doi.org/10.1016/j.cageo.2021.104864.

McCarthy, D.D., 1996. IERS Conventions (1996) (IERS Technical Note 21) Paris: Central Bureau of IERS - Observatoire de Paris, 1996. https://www.iers.org/IERS/EN/Publications/TechnicalNotes/tn21.html.

Mendes, V.B., Pavlis, E.C., 2004. High-accuracy zenith delay prediction at optical wavelengths. GRL 31 (14), 2004. https://doi.org/10.1029/2004GL020308.

Meyer, U., 2022. COST-G extended model covering GRACE+GRACE-FO period. Personal Commun.

Milani, A., Nobili, A.M., Farinella, P., 1987. In: Non-Gravitational Perturbations and Satellite Geodesy. Adam Hilger Ltd., Bristol, UK, p. 125 (ISBN 0-85274-538-9).

Montenbruck, O., Gill, E., 2012. In: Satellite Orbits; Model, Methods and Applications. Springer-Verlag, Berlin, Heidelberg. https://doi.org/10.1007/978-3-642-58351-3 (ISBN 978-3-540-67280-7).

Montenbruck, O., Hackel, S., Jäggi, A., 2018. Precise orbit determination of the Sentinel-3A altimetry satellite using ambiguity-fixed GPS carrier phase observations. J. Geod. 92, 711–726. https://doi.org/10.1007/s00190-017-1090-2.

Montenbruck, O., Hackel, S., Wermuth, M., Zangerl, F., 2021. Sentinel-6A precise orbit determination using a combined GPS/Galileo receiver. J. Geod. 95, 109. https://doi.org/10.1007/s00190-021-01563-z.

NASA, 2023. GEODYN Software Package. Available at https://space-geodesy.nasa.gov/techniques/tools/GEODYN/GEODYN.html. (Accessed 27 March 2026).

Neumayer, K.H., Schreiner, P., König, R., Dahle, C., Glaser, S., Mammadaliyev, N., Flechtner, F., 2024. EPOS-OC, a universal software tool for satellite geodesy at GFZ. In: Freymueller, J.T., Sánchez, L. (Eds.), Together Again for Geodesy. IUGG 2023. International Association of Geodesy Symposia. Springer, Cham. https://doi.org/10.1007/1345_2024_260.

Niell, A.E., 1996. Global mapping functions for the atmosphere delay at radio wavelengths. J. Geophys. Res. 101 (B2), 3227–3246 https://agupubs.onlinelibrary.wiley.com/doi/10.1029/95JB03048.

Pearlman, M.R., Noll, C.E., Pavlis, E.C., Lemoine, F.G., Combrink, L., Degnan, J.D., Kirchner, G., Schreiber, U., 2019. The ILRS: approaching 20 years and planning for the future. J. Geod. 93, 2161–2180. https://doi.org/10.1007/s00190-019-01241-1.

Peter, H., Berzosa, J., Fernández, J., Féménias, P., 2020. Long-term evaluation of estimated solar radiation pressure coefficients from Copernicus Sentinel-1, -2, -3 satellites, EGU General Assembly 2020, Online, 4–8 May 2020, EGU2020-5288. https://doi.org/10.5194/egusphere-egu2020-5288.

Peter, H., Meyer, U., Lasser, M., Jäggi, A., 2022. COST-G gravity field models for precise orbit determination of Low Earth Orbiting Satellites. Adv. Space Res. 69 (12), 4155–4168. https://doi.org/10.1016/j.asr.2015.06.019.

Petit, G., Luzum, B., 2010. IERS Conventions (2010) (IERS Technical Note 36) Frankfurt am Main: Verlag des Bundesamts für Kartographie und Geodäsie, p. 179. (ISBN 3-89888-989-6). https://www.iers.org/IERS/EN/Publications/TechnicalNotes/tn36.html.

Picone, J.M., Hedin, A.E., Drob, D.P., Aikin, A.C., 2002. NRLMSISE-00 empirical model of the atmosphere: statistical comparisons and scientific issues. J. Geophys. Res. 107 (A12), 1468. https://doi.org/10.1029/2002JA009430.

Priestley, K.J., Smith, G.L., Thomas, S., Cooper, D., Lee III, R.B., Walikainen, D., Hess, P., Szewczyk, Z.P., Wilson, R., 2011. Radiometric performance of the CERES Earth radiation budget climate record sensors on the EOS Aqua and Terra spacecraft through April 2007. J. Atmos. Ocean. Technol. 28 (1), 3–21. https://doi.org/10.1175/2010JTECHA1521.1.

Ray, J., Gurtner, W., 2010. RINEX Extensions to Handle Clock Information, version 3.02. https://files.igs.org/pub/data/format/rinex_clock302.txt.

Ray, J., Gurtner, W., Coleman, M.J., 2017. RINEX Extensions to Handle Clock Information, version 3.04. URL: https://files.igs.org/pub/data/format/rinex_clock304.txt.

Raymond, E.S., 2003. The Art of UNIX Programming. Addison-Wesley (ISBN:0-13-142901-9).

Rebischung, P., 2020. Switch to IGb14 reference frame, IGSMAIL-7921, 14 April 2020. https://lists.igs.org/pipermail/igsmail/2020/007917.html.

Rebischung, P., Schmid, R., 2016. IGS14/igs14.atx: a new framework for the IGS products. In: AGU Fall Meeting, p. G41A-0998.

Ricklefs, R. L., Moore, C.J., 2009. Consolidated Laser Ranging Data Format (CRD) v1.0. For the ILRS Data Formats and Procedures Working Group. https://ilrs.gsfc.nasa.gov/docs/2009/crd_v1.01.pdf.

Saastamoinen, J., 1972. Contributions to the theory of atmospheric refraction. Bull. Géod. 105, 279–298. https://doi.org/10.1007/BF02521844.

Schaer, S., 2018. SINEX BIAS – Solution (Software/technique) Independent EXchange Format for GNSS Biases Version 1.00. https://files.igs.org/pub/data/format/sinex_bias_100.pdf.

Shi, C., Zhao, Q., Geng, J., Lou, Y., Ge, M., Liu, J., 2008. Recent development of PANDA software in GNSS data processing. In: Proc. SPIE 7285, International Conference on Earth Observation Data Processing and Analysis (ICEODPA). https://doi.org/10.1117/12.816261.

Shihora, L., Balidakis, K., Dill, R., Dahle, C., Ghobadi-Far, K., Bonin, J., Dobslaw, H., 2022. Non-tidal background modeling for satellite gravimetry based on operational ECWMF and ERA5 reanalysis data: AOD1B RL07. J. Geophys. Res. Solid Earth 127 (8), e2022JB024360. https://doi.org/10.1029/2022JB024360. AOD1B RL07 files: ftp://isdcftp.gfz-potsdam.de/grace-fo/Level-1B/GFZ/AOD/RL07.

Springer, T., Dilssner, F., Escobar, D., Flohrer, C., Otten, M., Svehla, D., Zandbergen, R., 2011. NAPEOS: The ESA/ESOC Tool for Space Geodesy. Geophys. Res. Abstr. 13, EGU2011-8287. Available at https://meetingorganizer.copernicus.org/EGU2011/EGU2011-8287.pdf. (Accessed 27 March 2026).

Steigenberger, P., Thoelert, S., Montenbruck, O., 2018. GNSS satellite transmit power and its impact on orbit determination. J. Geod. 92, 609–624. https://doi.org/10.1007/s00190-017-1082-2.

Vallado, D.A., 2013. Fundamentals of Astrodynamics and Applications, fourth ed. Space Technology Library (ISBN: 978-1881883180).

Villiger, A., Dach, R., Arnold, D., Schaer, S., Selmke, I., Kalarus, M., Prange, L., Stebler, P., Jäggi, A. 2020: CODE repro3-IGb14 product series. Published by Astronomical Institute, University of Bern. http://www.aiub.unibe.ch/download/REPRO_I20; https://doi.org/10.7892/boris.135946.

Villiger, A., 2022. Upcoming switch to IGS20/igs20.atx and repro3 standards. IGSMAIL-8238, 26 July 2022. Available at https://lists.igs.org/pipermail/igsmail/2022/008234.html. (Accessed 27 March 2026).

Wermuth, M., Montenbruck, O., Van Helleputte, T., 2010. GPS high precision orbit determination software tools (GHOST), in: 4th international conference on astrodynamics tools and techniques, Madrid, Spain, 3–6, Citeseer, 2010. https://elib.dlr.de/74046/1/Wermuth_et_al%282010%29.pdf. (Accessed 27 March 2026).

Wu, J.T., Wu, S.C., Hajj, G., Bertiger, W.I., Lichten, S.M., 1993. Effects of antenna orientation on GPS carrier phase. Manuscr. Geod. 18, 91–98. https://doi.org/10.1007/BF03655303.