

G-NUT/TEFNUT

Version 3 (4)

User manual

Pavel VÁCLAVOVIC and Jan DOUŠA
email: support@gnutsoftware.com

Contents

1	Introduction	3
1.1	Purpose	3
1.2	Versions	3
1.3	Requirements	4
1.4	Licences	5
1.5	Installation	5
1.6	Development and support	5
1.7	Acknowledgements	5
2	Precise Point Positioning	6
3	Using the software	7
3.1	User help	7
3.2	Running modes	8
3.3	Multiple inputs/outputs	8
4	Configuration	10
4.1	XML settings structure	11
4.2	General configuration	11
4.3	GNSS configuration	12
4.4	Navigation data configuration	13
4.5	Input configuration	14
4.6	Output configuration	15
4.7	Processing configuration	17
4.8	Site metadata configuration	23
5	Processing modes	25
5.1	Post-processing mode	25
5.2	Real-time mode	25
5.3	Parallel vs. sequential processing	26
5.4	Handling station/receiver names	26

6	Output formats	28
6.1	Log outputs	29
6.2	Solution outputs	31
6.3	Troposphere outputs	32
7	User console	34
7.1	Data container browsing	35
7.2	Data decoder monitoring	36
7.3	Repetitive command call	37
7.4	Application re-configuration	37
	References	38

1. Introduction

1.1 Purpose

The G-Nut/Tefnut is a software designed for estimating tropospheric parameters, such as zenith total delay (ZTD), horizontal gradients, and total slant delays, on the basis of analysing Global Navigation Satellite Systems (GNSS) data. GNSS data are processed using the Precise Point Positioning (PPP) method [1] supported with the Kalman forward filter. Therefore, estimates at one station are gained with a high temporal resolution in real-time as well as in a post-processing mode. In case of post-processing, results from the Kalman filter can be recalculated and improved with an application of the backward smoothing algorithm [2].

The software supports these input formats (including Hatanaka's CRX, gz and Z compression) which documentations can be found at [3]:

- **Data files:**

- RINEX 2/3/4 and BNC OBS (observation data),
- RINEX 2/3/4 (navigation data),
- RINEX 2/3 (clock corrections),
- SP3 (precise ephemeris),
- ERP (earth rotation parameters),
- BLQ (ocean tide modelling),
- ORBEX (satellites attitudes),
- SINEX (metadata),
- ANTEX (antenna phase center models),
- BIAS SINEX (signal biases).

- **Data streams:**

- RTCM 3 (observation, navigation, SSR data, [4]),
- BNC OBS (observation data decoded by BKG Ntrip client, [5]),
- BNC RTCM (SSR corrections decoded by BKG Ntrip client, [5]).

The software supports these output file formats (including gz and Z compression):

- TROPO SINEX 1/2 (tropospheric parameters),
- G-Nut's data product formats,
- Log files.

The software can be executed from the console user interface (CUI). It can be configured using an XML file, from a command-line arguments, or both. It has no graphic user interface (GUI).

The software is currently available for Linux, Windows and MacOS operating systems.

1.2 Versions

*Important note: individual functionalities are not always explicitly distinguished within this manual. The term **Tefnut** in examples below refers to a version relevant for the operation.*

The G-Nut/Tefnut is being released in three different software versions:

Tefnut PRO

A commercial version dedicated to a post-processing solution

- full configuration using XML file
- multi-GNSS solution
- estimates of tropospheric horizontal gradients
- retrievals of tropospheric slant delays
- backward smoothing
- outputs in SINEX_TRO 2
- high-resolution product
- multi-station parallel processing

Tefnut RT

A commercial version extending the functionality of the *Tefnut PRO* for the real-time operations

- estimate of tropospheric parameters in real time
- TCP and NTRIP clients
- monitoring via a software user console
- input RTCM data (binary) streams
- input BNC data (ASCII) formats for synchronized observations and precise products

Tefnut TRIAL

A time-limited trial version with a full functionality of the *Tefnut RT*.

1.3 Requirements

As mentioned in the section 2, precise satellite orbits and clocks corrections are the mandatory inputs for the processing. The software can read such products from files in different formats or from a stream using the RTCM or BNC formats. Satellite biases represent optional inputs, however, are recommended for improved precision, in particular when supporting integer ambiguity resolution. The code biases are not important when processing signals also used for satellite clock corrections estimates. Such a standard has been defined within the International GNSS Service (IGS, [6]), so it can be considered when using products consistent with IGS. Additional inputs for different models are also not mandatory, and users should consider them on their own responsibility.

As far as hardware requirements are concerned, the only limitation consists in memory resources. The memory consumption can be presented with an example of real-time processing of 230 stations with 10 seconds sampling rate when results are written to output files on an hourly basis. In such a case, approximately 11 MB is allocated.

1.4 Licences

The Tefnut PRO and Tefnut RT are provided under a commercial licence, for details see <https://www.gnutsoftware.com>.

The Tefnut Trial is a trial version for a limited period of time.

1.5 Installation

The software is distributed from <http://www.gnutsoftware.com>. It can be downloaded as a pre-compiled binary (static/dynamic, 32b/64b), and it does not need any specific installation or a compilation procedure. A static version is provided for a flexible support of various Linux distributions/releases, however, the shared Linux version is strongly recommended for any real-time applications.

1.6 Development and support

The software has been developed using the G-Nut core library [7] initially designed at the Geodetic Observatory Pecný of the Research Institute of Geodesy, Topography and Cartography. Since 2018, both the library and user applications has been maintained and further developed only by the G-Nut Software s.r.o..

In case of a bug reporting, a feedback or a wish-list, contact us via the email: support@gnutsoftware.com.

Remark: support on usage is provided on the best-effort basis only. In case you need any specific user support, contact us for negotiation.

1.7 Acknowledgements

We acknowledge the availability of third-party libraries used within the G-Nut:

- Newmat - C++ Matrix library (http://www.robertnz.net/nm_intro.htm).
- Pugixml - a light-weight, simple and fast XML parser for C++ (<http://pugixml.org/>).
- RNXCMP - a Hatanaka's RINEX compression (<https://terras.gsi.go.jp/ja/crx2rnex/>) [8].
- md5 - a reference implementation of RFC 1321 converted to C++ by Frank Thilo for bzflag (<http://www.bzflag.org>).
- rapidjson - a fast JSON parser/generator for C++ with both SAX/DOM style API (<https://rapidjson.org/>).
- zlib - a free lossless data-compression C library (<https://www.zlib.net>).
- gzstream - a small C++ wrapper providing the functionality of the zlib C-library (<https://www.cs.unc.edu/Research/compgeom/gzstream/>).

2. Precise Point Positioning

The Precise Point Positioning (PPP) [1] method is implemented in the G-Nut/Tefnut software using undifferenced observations from a single receiver, or a group of receivers. Any systematic errors influencing GNSS measurements are thus neither eliminated nor reduced.

The errors can be separated into four categories: satellite induced biases, receiver induced biases, atmospheric effects, and environmental conditions. All of them must be either known from external products, estimated within the adjustment process, or eliminated via combining signals transmitted at different frequencies.

Precise satellite orbit and clock products are thus essential inputs for the PPP method. It also requires precise models of all the errors in the absolute sense as these are not eliminated by a differencing operator. Even hardware biases are not eliminated and need to be introduced or estimated. While satellite code biases are usually introduced from external sources, receiver code biases are estimated together with the receiver clock offset. Satellite and receiver phase biases are then commonly absorbed by initial phase ambiguities when estimated as float numbers. A disadvantage of the PPP is a long convergence period (tens of minutes) needed for an accurate estimation (decorrelation) of all parameters. This initial convergence particularly concerns of a solution using initial phase ambiguities in terms of float numbers, i.e. not resolved to integer values suggested by their natural characteristics. The ambiguity resolution is available only if satellite phase biases are well known from an external product.

The software supports these external products:

- precise satellite orbits
- precise satellite clocks
- satellite biases

The models implemented in the software are

- closed formula models:
 - solid earth tides
 - phase wind-up
 - relativistic effects on satellite clocks
- models using additional input data:
 - ocean tide loading
 - pole tides
 - receiver and transmitter antenna phase center offsets and variations

The list of parameters estimated by the software consists of:

- receiver coordinates
- receiver clock corrections
- zenith wet delays
- horizontal tropospheric gradients
- inter-system biases (for each individual GNSS, but GPS)
- initial ambiguities

3. Using the software

This chapter shows how to execute the software and to introduce command-line arguments.

3.1 User help

A basic help (*-help*) shows principal command-line arguments.

```
G-Nut/Tefnut Pro [3.1] compiled: Jan 11 2021 18:41:07 (Rev: 3113)

Usage:

  -h|--help          .. this help
  -V                 .. software version
  -x file            .. configuration input file
  -X                 .. configuration stdout (default)
  -Z                 .. log-file without time stamps
  -z file            .. configuration output file
  xml-settings       .. XML settings via a command line (see below)

XML settings via command line:

:elem:subelem val    .. set xml element value (e.g. :inputs:rinexo <rinexo file>)
:elem:attribute=val  .. set xml attribute value (e.g. :outputs:verb=1)
:elem[ID]:subelem val .. set xml element value (e.g. :sys[GPS]:sat "G01_G02_G03")
:elem[ID]:attribute=val .. set xml attribute value (e.g. :site[GOPE]:rec="RECEIVER")
```

An extended help (*-X*) then gives a detailed list of applicable configuration options including a brief description and default values.

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<!DOCTYPE config>
<config>
  <!--general settings:
  max_cores=integer      .. max cores to be used (0=any)
  max_files=integer      .. max files in a batch (0=all)

  beg    %Y-%m-%d %H:%M:%S .. beg time (empty=auto)
  end    %Y-%m-%d %H:%M:%S .. end time (empty=auto)
  int    integer           .. data sampling interval (empty=all)
  rec    list of strings   .. receiver(s) (empty=all)
  sys    list of strings   .. GNSS system(s) (empty=all)
  -->
  <gen max_cores="1" max_files="1"><beg /><end /><int /><rec /><sys /></gen>

  <proc>
  <!--filter settings:
  flt_method=enum        .. type of filtering method (SRCF, Kalman)
  smt_method=enum        .. type of smoothing method (RTS, RTSSVD, NONE)
  clk_white=float        .. white noise for clocks [m]
  crd_white=float        .. white noise for coordinates [m]
  ztd_rwalk=float        .. random walk process for ZTD [mm/sqrt(hour)]
  grd_rwalk=float        .. random walk process for GRD [mm/sqrt(hour)]
  ...
  -->
  <!--parameters settings:
  trop=enum              .. tropospheric parameters (NONE, ZTD, ZTD+GRD, ZTD+GRD+SLT)
  coor=enum              .. coordinates parameters (EST, FIX, KIN)

  crd_sigma=float        .. accuracy of initial coordinates [m]
  amb_sigma=float        .. accuracy of initial ambiguity [m]
  ztd_sigma=float        .. accuracy of initial zenith path delay [m]
  grd_sigma=float        .. accuracy of initial tropo gradients [m]
  ...
  -->
```



```

<!--models settings:
ztd=enum          .. tropospheric model [SAASTAMOINEN, GPT,...]
ztd_mf=enum       .. tropospheric mapping function [COSZ, GMF, ...]
grd_mf=enum       .. tropo gradient mapping function [TILTING, CHEN_HERRING, BAR_SEVERER ...]
attitude=enum     .. satellite attitude model [NOMINAL, RTCM, STEER, QUATERNIONS, ZERO]
eclipse=enum      .. use eclipsing satellites [USE, SKIP]
windup=enum       .. use windup correction [USE, NONE]
pco_corr=enum     .. use PCO corrections [ALL, RECEIVER, TRANSMITTER, NONE]
pcv_corr=enum     .. use PCV corrections [ALL, RECEIVER, TRANSMITTER, NONE]
...
-->
.....
</config>

```

3.2 Running modes

The software can run in two different modes: offline and real-time. While all input files are read only once in case of the offline processing, an infinite loop is used for reading data from input data streams in case of real-time processing. Offline is the default mode, while real-time mode is activated with the attribute *run_rtime* of the element `<gen>`, see Section 4.2. Also a short command-line argument *-rtime* can be used instead. Real-time additionally requires the setting of a process delay *run_delay* in the same element.

If the mandatory inputs are neither available nor configured, the software exits with an error message in the standard error output.

3.3 Multiple inputs/outputs

Multiple inputs for a single format, e.g. RINEX files, can be easily introduced in Linux by creating file lists using shell commands:

```

Tefnut :inp:rinexo "$(ls_dir/2022/100/*2022100*crx.gz)" # site expansion
Tefnut :inp:rinexo "$(ls_dir/2022/*/GOPE00CZE*crx.gz)" # time expansion

```

However, the software can also introduce a large number of input files (and/or data streams) by using an internal software mask with pre-defined variables supporting time/site expansions of individual configurations. The following example shows an utilization of multiple files when using a time/site-specific input mask defined for three stations in July, 2022:

```

Tefnut :gen:rec 'GOPE00CZE_GOP600CZE_GOP700CZE' \
:gen:beg '2022-08-01T00:00:00' \
:gen:end '2022-08-31T00:00:00' \
:inp:rinexo 'dir1/(REC9)_R_%Y%j0000_01D_30S_M0.crx.gz' \
:inp:rinexn 'dir2/BRDC00GOP_R_%Y%j0000_01D_MN.rnx.gz' \
:inp:rinexc 'dir4/igs%W%w.clk_30s' \
:inp:sp3 'dir3/igs%W%w.sp3'

```

Additional notes:

- the `<beg>/<end>` elements have to always be defined when using the time expansion,
- the `<rec>` element has to always be defined when using the site expansion,
- mask can be specified for any input (output) format,
- mask can be applied in XML configuration or command-line arguments, see Section 4,
- full description of the configuration is provided in Section 4.

Any generic site mask found in the configuration is expanded to a variable (*rec*) and (*REC*) for the lower-case and the upper-case sitename, respectively. Additionally, user may request 4-char and 9-char (upper/lower) sitename by writing (*REC4/rec4*) and (*REC9/rec9*), respectively.

The time variables listed bellow are eventually substituted for a reference epoch (or a sequence of reference epochs) of the processed data:

- '%Y' – 4-char year,
- '%y' – 2-char year,
- '%j' – doy of year (DoY),
- '%m' – month of year,
- '%d' – day of month,
- '%w' – day of GPS week,
- '%W' – GPS week,
- '%H' – hours of day,
- '%M' – minutes of day,
- '%S' – seconds of day.

If a site-/time-dependent variable is found within input (output) path, it is considered as a generic file mask instead of a single filename. Such mask is then expanded into a site-/time-specific list of input (output) candidate files that will be further checked. Only those relevant (existing for input and applicable for output) are eventually read (written), otherwise ignored.

Remark: the time-dependent and site-dependent masks are expanded only when the processing interval and the list of stations are configured, respectively.

Finally, specific setting attributes may control efficient operation on multiple inputs, namely 1) parallel or sequential file processing, 2) limit numbers of cores, files, and sites used within a single processing batch. The following attributes of the element <gen> define

- *max_cores* - a maximum number of cores to be used (0=auto),
- *max_files* - a maximum number of files to be used (0=auto),
- *max_sites* - a maximum number of sites to be used (0=auto).

4. Configuration

The software can be configured from the XML configuration file, using command-line arguments, or mixing both. The XML configuration principally represents an internal software structure for user settings. Command-line arguments additionally enable user a) to configure completely the software, or b) to modify default settings or the XML configuration for a particular user execution.

This section describes only the XML configuration since any command-line argument expresses only a conversion to the XML settings when applying the following principles:

```
:elem:subelem val .. set xml element value (e.g. :inp:rinexo <rinexo file>)
:elem:attribute=val .. set xml attribute value (e.g. :out:verb=1)
:elem[ID]:subelem val .. set xml element value (e.g. :sys[GPS]:sat "G01_G02_G03")
:elem[ID]:attribute=val .. set xml attribute value (e.g. :site[GOPE]:rec="RECEIVER")
```

Contrary to the XML settings, there are constraints that need to be considered when using command-line arguments:

- duplicitous argument (e.g. :elem:attribute="value") overwrites default or existing one,
- multi-elements (:sys[id], :site[id]) require an element identifier (e.g. [GPS], [GOPE]).

It is important to note that for avoiding duplicitous command-line arguments, user has to insert multiple values in quotation marks. Hence, any input previously configured either from the XML or command-line settings will be replaced with the latest command-line argument.

A basic help (*-help*) and a full help (*-X*) has already been introduced in Section 3.1.

Example below shows software executions when using the XML settings

```
Tefnut -x my_config.xml
```

and when using the command-line settings

```
Tefnut :inp:rinexo GOPE1730.17o \
:inp:rinexn BRDC1730.17p \
:out:tef GOPE1730.tef \
:out:log GOPE1730.log \
:out:verb=1 \
:gen:beg "2017-06-22_00:00:00" \
:gen:end "2017-06-23_00:00:00" \
:gen:sys "GPS_GLO" \
:gen:int 30
```

Mixing both configurations can be handfull when specifying some static (user-default) parameters in the XML file (e.g. generic configuration, output file masks, site metadata) and when adding/modifying selected (dynamic) parameters from the command-line (e.g. input files):

```
Tefnut -x metadata.xml :inp:rinexo "$(ls_data/????????_R????????_01D_30S_MO.crx.gz)" \
:inp:rinexn "$(ls_brdc/BRDC????_R????????_01D_MO.rnx.gz)"
```

*Hint: in this particular case, it is recommended to store actual configuration (*-z final.cfg*) along with archiving output files.*

4.1 XML settings structure

A structure of the Tefnut XML configuration file consists of elements and attributes of the Extensible Markup Language (XML). The configuration is placed within a root element `<config>` and following XML tree of child elements:

- `<gen >` for general settings,
- `<sys >` for GNSS-specific settings,
- `<nav >` for navigation data settings,
- `<inp >` for input settings,
- `<out >` for output settings,
- `<proc >` for PPP processing settings,
- `<site >` for site metadata settings.

Selected elements can be omitted (default values are considered then), however, input and output elements are not predefined, and thus mandatory. Individual elements are described in following sections.

4.2 General configuration

The element `<gen>` includes general settings for the software operation such as a start and end time of a processing period, requested GNSS constellations, list of GNSS sites (receivers), and a sampling interval of the processing. The settings is defined using the XML sub-elements.

```

<gen
  max_cores="0"           # maximum number of cores to be used (0=all)
  max_sites="0"          # maximum number of sites in a batch (0=all)
  max_files="0"          # maximum number of files in a batch (0=all)

  run_rtime="false"      # boolean to activate real-time mode (default: false)
  run_delay="0"          # delay[s] of processing real-time data (default:0 s)
>
  <rec>  GOPE POTS WTZR   </rec>
  <sys>  GPS GLO GAL BDS QZS </sys>
  <int>  300              </int>
  <beg>  "2014-03-01_00:00:00" </beg>
  <end>  "2014-03-01_23:59:59" </end>
</gen>

```

Remark: the general settings is applied at the decoding level, thus data/products are filtered out by the settings prior any data analysis.

The attribute ***max_cores*** defines a maximum number of cores selected for the processing (1 is sequential, 0 and > 1 means parallel) which has already been described in Section 3.3. The attribute ***max_files*** and ***max_sites*** specifies then a maximum number of files and sites, respectively, used in a single processing batch.

The attribute ***run_rtime*** activates the software for real time. It can be set to boolean while the default is 'false' (Section 3.2). The attribute ***run_delay*** defines a delay [sec] of the real-time data processing.

The `<rec>` element specifies a list of receivers (stations) to be processed. The list may be left empty (default), considering all the sites identified from input data or station meta data configuration can be processed. More information can be found in Section 5.4.

The `<sys>` element specifies global and regional systems selected for the processing. These 3-char GNSS abbreviations are allowed: **GPS**, **GLO**, **GAL**, **BDS**, **QZS**, **SBS**, and **IRN** for GPS, GLONASS, Galileo, BeiDou, QZSS, SBAS, and IRNSS, respectively. An empty element (default) enables all GNSS and regional systems.

Remark: in empty element the '-' (minus) pre-fix enables rejecting any particular system.

The `<int>` element sets the sampling rate [*sec*] of the processing which is used to filter out any input data during the decoding process.

Remark: the <int> element contains usually an integer number, however, for processing of high-rate data (> 1Hz), a float number may also be set as the sampling interval with extra decimals indicating a time resolution of original input. For example, setting 0.1 means a 10-Hz processing of 10Hz input data, while 0.10 requests a 10Hz processing of original 100Hz data.

The elements `<beg>` and `<end>` define the processing start and the end, respectively, both in the format `'%Y-%m-%d %H:%M:%S'`. The format allowing to merge date and time with the 'T' character is acceptable too, i.e. `'%Y-%m-%dT%H:%M:%S'`. If configured, both elements define the period to expand time variables using the input/output update rate, see Section 3.3. If these elements are left empty, all observations are processed and input/output file mask cannot be used.

4.3 GNSS configuration

Constellation-specific configuration is enabled in the `<sys>` element, which needs to be completed with a mandatory *id* attribute in order to distinguish individual GNSS constellations, see Section 4.2. Hence, multiple elements are supported.

Several other attributes are enabled to configure generic characteristics of individual systems:

- *sigma_c* – a priori accuracy of code observations [m]
- *sigma_l* – a priori accuracy of carrier-phase observations [m]
- *maxres_c* – maximum code residual for outlier detection [m]
- *maxres_l* – maximum carrier-phase residual for outlier detection [m]
- *slip_thr* – threshold for cycle slip detection [cycles]

Each system may also contain the following sub-elements for selecting:

- `< nav >` for navigation types,
- `< sat >` for satellites,
- `< type >` for observation types,
- `< band >` for observation frequency bands,
- `< attr >` for observation tracking attributes,
- `< gobs >` for observation signal-specific 3-char codes (direct or add-on signal settings).

These can be used for defining a set of conditions applied for input data filtering during the decoding process. By default, all the settings are empty, thus no filtering is applied. By introducing any single value into any element then activates corresponding filter. The settings for the `<type>`, `<band>`, and `<attr>` elements are binded, consequently, all values should be specified together. The `<gobs>` eventually enables extending the type, frequency and tracking settings with an explicit specification of signal(s) by corresponding RINEX standards codes.

An example of settings for Galileo (not very meaningful) is given below:

```
<sys id='GAL'>
  <sat>  E01 E05 E06 E08 E09 E12 E14  </sat>  # disable unspecified Galileo satellites
  <nav>  FNAV INAV_E07                </nav>  # disable INAV_E01
  <type>  C L                          </type>  # disable Doppler and signal-to-noise ratio
  <band>  1 2 5 7 8                    </band>  # disable E6 frequency band observations
  <attr>  C L X P W                    </attr>  # disable unspecified observation attributes
  <gobs>  C1C C2P L1C L2P S1C S2P L2P </attr>  # add-on specifically mentioned signals
</sys>
```

The `<sat>` element specifies a list of selected satellites. If the element is not specified in the configuration file (or is empty), all available (supported) satellites for a given constellation is used by default.

Remark: similarly as for the `<sys>`, the '-' (minus) pre-fix enables a negative selection, i.e. rejecting only particular satellite in the `<sat>` element.

The `<nav>` element specifies the navigation data types when (currently) supporting:

- **FNAV** for Galileo FNAV from the E5a signal,
- **INAV** for Galileo INAV from any signal,
- **INAV_E01** for Galileo INAV from the E1 signal,
- **INAV_E07** for Galileo INAV from the E5b signal,
- **NAV** for any navigation type.

4.4 Navigation data configuration

An optional handling of navigation data can be used for two specific purposes:

1. filtering, checking, and handling navigation data within the software, and
2. generating merged and quality control navigation data, i.e. BRDC file, [9]).

Selected parameters can be used in any G-Nut application:

- **chk_record** (boolean) – range checking of individual navigation records
- **chk_health** (boolean) – filtering out satellites with unhealthy status
- **chk_duplic** (boolean) – filtering out duplicitous navigation records
- **chk_tot** (boolean) – checking navigation data validity with respect to the ToT epoch
- **key_tot** (boolean) – storing/searching navigation records according to the ToT epoch
- **ext_nav** (double) – extending navigation data validity interval

Principally, the options may affect searching navigation data for a user requested epoch in the processing (i.e. depends on how important it is for the specific use case).

The attribute ***chk_record*** activates a simple range checking control of navigation data (e.g. time validity, realistic value, obvious error).

The attribute ***chk_health*** and ***chk_duplic*** is used to remove (from the processing) all actually unhealthy satellites and duplitious navigation data for a particular satellite and transmission time, respectively.

The attribute ***chk_tot*** enables checking the Time of Transmission (ToT) with respect to the Time of Clocks (ToC) when searching relevant navigation data for a user requested epoch. Navigation record is ignored whenever the ToT is later the ToC which indicates data re-transmission. This is important for simulated real-time mode considering navigation data actually available.

The attribute ***key_tot*** defines using the ToT as an internal key for sorting navigation data (by default the ToC is used).

The attribute ***ext_nav*** enables setting a multiplier for changing a validity period of individual navigation records with respect to the nominal one which is specific for each GNSS.

The default settings should be convenient for a majority of PPP strategies:

```
<nav chk_record="true"  chk_health="true"  chk_duplic="true"  chk_tot="true"  key_tot="true"  ext_nav="1.0" />
```

4.5 Input configuration

Supported input formats were listed in Section 1.1. The `<inp>` element includes sub-elements for inputs of observation data, navigation data, metadata, precise products and models. Each sub-element represents a specific format which may contain a space-separated list of more input paths, e.g.:

```
<inp>
  <!-- input settings: path may include prefix file:// [..tcp,ntrip]
  <rinexn>    path1 [path2 ..]  </rinexn>    # for RINEXN    format (file)
  <rinexo>    path1 [path2 ..]  </rinexo>    # for RINEXO    format (file)
  <rinexc>    path1 [path2 ..]  </rinexc>    # for RINEXC    format (file)
  <sp3>       path1 [path2 ..]  </sp3>       # for SP3       format (file)
  <sinex>     path1 [path2 ..]  </sinex>     # for SINEX     format (file)
  <atx>       path1 [path2 ..]  </atx>       # for ANTEX     format (file)
  <erp>       path1 [path2 ..]  </erp>       # for ERP       format (file)
  <blq>       path1 [path2 ..]  </blq>       # for BLQ       format (file)
  <orbex>     path1 [path2 ..]  </orbex>     # for ORBEX     format (file)
  <biasinex>  path1 [path2 ..]  </biasinex>  # for BIAS SINEX format (file)
  <rtcm>      path1 [path2 ..]  </rtcm>      # for RTCM      format (file or stream)
  <bncobs>    path1 [path2 ..]  </bncobs>    # for BNC OBS   format (file or stream)
  <bncrtcm>   path1 [path2 ..]  </bncrtcm>   # for BNC SSR   format (file or stream)
  -->
</inp>
```

The path represents a generic input such as data files, data streams, and file masks.

Remark: the file mask is not a valid path, but contains user- or time-specific variables going to be replaced by valid strings prior a file searching.

The path may include one of the following structures (the ***file://*** prefix is optional):

- **file://filename.ext** – file accessible within a local/network file system,
- **tcp://host:port** – TCP data stream from host+port (Tefnut-RT only),
- **ntrip://mnt/user:pwd@host:port** – NTRIP data stream from mountpoint (mnt), accessible with username (user) with password (pwd) from a host+port (Tefnut-RT only).

The RINEX format version is automatically recognised in the software input as well as files with the **Hatanaka** compression. Files are also automatically compressed/decompressed by the software when using the **.gz/.Z** extension in the file name.

Remark: the file name doesn't play any role in the format specification.

An example below shows a mixed input for GNSS observation and navigation files using relative and absolute paths and RINEX 2 and 3 input formats:

```
<inp>
  <rinexn> file://../data/brdc/brdm0220.15p.gz    </rinexn>
  <rinexo> file://../data/obs/onsa0220.15o.gz
           /home/jan/data/obs/GOPE00CZE_R_201522000000_01D_30S_M0.crx.gz
</rinexo>
</inp>
```

Remark: Windows users should specify the file path as supported by the platform.

4.6 Output configuration

Supported output formats were also listed in Section 1.1. The `<out>` element includes sub-elements for individual outputs of the PPP processing and for log files. Similarly as for inputs, each sub-element represents a specific format, but contains only a single output path or file mask using site/time variables.

```
<out
  verb="[integer]"                # log verbosity (default: 0)
  append="[boolean]"              # log appending (default: false)
>
  <!-- output settings: path may include prefix file://
  <log>      path </log>           # for LOG format      (file)
  <ppp>      path </ppp>          # for PPP LOG format (file)
  <tef>      path </tef>         # for TEF format     (file)
  <trosinex> path </trosinex>     # for TroSINEX 2 format (file)
  <trosinex0> path </trosinex0>  # for TroSINEX 1 format (file)
  -->
</out>
```

The `<out>` element contains two attributes relevant for the log configuration:

- **verb** [0-9] defines a verbosity level representing details of printed information.
- **append** (boolean) enforces a log file continuation in real-time processing. The 'false' is default and means creating a new or overwriting an existing file.

Remark: if the path contains non-existing directories, these will be created recursively.

The example below shows a simple output settings of the Tefnut processing:

```
<out verb=2 append="false">
  <log>      OUT/tefnut-2015-220.log.gz    </log>
  <tef>      OUT/(REC)-2015-220.tef.gz    </tef>
  <trosinex> OUT/tefnut-2015-220.tro.gz   </trosinex>
</out>
```


Remark: any output file, in particular a log file, may also be redirected in Linux to the standard and error outputs (/dev/stdout, /dev/stderr).

The native `<tef>` format for estimated parameters is described in Section 6. Note the site variable (*REC*) enabling a mask definition for a common output for all simultaneously processed files/stations. In real-time processing, also time variables within the file mask specification are further needed for a regular file generation, i.e.

```
<out verb=2 append="false">
<log>      OUT/tefnut-%Y%m%d-%H%M.log.gz      </log>
<tef>      OUT/(REC)-%Y%m%d-%H%M.tef.gz      </tef>
<trosinex> OUT/tefnut-%Y%m%d-%H%M.tro.gz     </trosinex>
</out>
```

A corresponding file update is defined a) implicitly, from the shortest range of any time variable found within the filename (every minute in the above case) or b) explicitly, by a user setting via the attribute *upd* [sec] of the given format element (every hour in the below case). Similarly, an output file length and output data sampling rate can be set using the attribute *len* [sec] and *smp* [sec], respectively. For details, see Section 6.

```
<out verb=2 append="false">
  <trosinex upd=3600 len=3600 smp=300> OUT/tefnut-%Y%m%d-%H%M.tro.gz  </trosinex>
</out>
```

The Tefnut log file reports line-by-line output prefaced with an event epoch (system time), followed with an object identifier (the one throwing the message) and its verbosity, and the message itself. An example of a log file with *verb=2* is given below:

```
2022-06-20 22:27:35 [gsetbase:2] xconfig: read from file conf/geb.cfg
2022-06-20 22:27:35 [gsetbase:0] program: G-Nut/Tefnut RT [3.3] Jun 20 2022 21:54:29 (Rev: 3696M)
2022-06-20 22:27:35 [gsetbase:0] =====
2022-06-20 22:27:35 [gsetbase:1] setting: conf/tef.cfg
2022-06-20 22:27:35 [gsetbase:1] logfile: log/gnut.log
2022-06-20 22:27:35 [gsetbase:1] verbose: 2
2022-06-20 22:27:35 [gsetbase:1] =====
2022-06-20 22:27:35 [atx:2] VERS: 1.4
2022-06-20 22:27:35 [atx:2] reading PCV TYPE / REFANT
2022-06-20 22:27:35 [atx:2] reading COMMENT
2022-06-20 22:27:35 [atx:2] reading END OF HEADER
2022-06-20 22:27:35 [gallpcv:2] add PCV G01 1992-11-22 00:00:00 2008-10-16 23:59:59
2022-06-20 22:27:35 [atx:2] add pcv 1992-11-22 00:00:00 2008-10-16 23:59:59 BLOCK IIA
2022-06-20 22:27:35 [gallpcv:2] add PCV G01 2008-10-23 00:00:00 2009-01-06 23:59:59
2022-06-20 22:27:35 [atx:2] add pcv 2008-10-23 00:00:00 2009-01-06 23:59:59 BLOCK IIA
2022-06-20 22:27:35 [gallpcv:2] add PCV G01 2009-03-24 00:00:00 2011-05-06 23:59:59
2022-06-20 22:27:35 [atx:2] add pcv 2009-03-24 00:00:00 2011-05-06 23:59:59 BLOCK IIR-M
...
2022-06-20 22:28:10 [gallpcv:2] add PCV TWIVP6050_CONE NONE 1980-01-01 00:00:00 2079-12-31 23:59:59
2022-06-20 22:28:10 [atx:2] add pcv 1980-01-01 00:00:00 2079-12-31 23:59:59TWIVP6050_CONE NONE
2022-06-20 22:28:11 [gallpcv:2] add PCV TWIVSP6037L NONE 1980-01-01 00:00:00 2079-12-31 23:59:59
2022-06-20 22:28:11 [atx:2] add pcv 1980-01-01 00:00:00 2079-12-31 23:59:59TWIVSP6037L NONE
2022-06-20 22:28:11 [gserv_ios:0] READ ATX [-]: gen/igs14_Dec21.atx completed [21407207] 29.905 sec
2022-06-20 22:28:11 [rinexc:2] VERS: 2.00
2022-06-20 22:28:11 [rinexc:2] PGM / RUN BY / DATE
2022-06-20 22:28:11 [rinexc:2] TYPES OF DATA
2022-06-20 22:28:11 [rinexc:2] END OF HEADER
2022-06-20 22:28:13 [gserv_ios:0] READ RINEXC [-]: data/cnt21923.clk completed [6397753] 1.902 sec
2022-06-20 22:28:13 [sp3:2] add_data : OBJ [OBJ]
2022-06-20 22:28:13 [sp3:2] add_data : PREC [ORB]
2022-06-20 22:28:13 [sp3:0] READ sp3: beg:2022-01-12 00:00:00 epo:288 acr:CNES sys:IGS14 flg:P dat:ORBIT orb:IGU
2022-06-20 22:28:13 [sp3:2] READ sample: 300
...
2022-06-20 22:28:20 [gallobj:2] object: GOP6 not found.
2022-06-20 22:28:20 [gallobj:2] ADD new obj GOP6
2022-06-20 22:28:20 [rinexc:2] Object created, using RINEX header: GOP6 2022-01-12 01:00:00
2022-06-20 22:28:26 [gserv_ios:0] READ RINEX0 [-]: data/GOP6012b.22o completed [27542722] 5.084 sec
2022-06-20 22:28:26 [rinexo:2] data/GOP6012b.22o [filtered data] xBEG:0 xEND:0 xSMP:0 xSYS:93961
2022-06-20 22:28:26 [rinexo:0] Warning: OBSERVER/AGENCY not available! .. file://GOP6012b.22o
2022-06-20 22:28:26 [gserv_ppp:2] ADD site GOP6 to PPP process
2022-06-20 22:28:26 [main:0] PP filter started
...
2022-06-20 22:32:28 [main:0] PP filter finished
2022-06-20 22:32:29 [gserv_ios:0] SAVE FLT [-]: out/GOP6.tef completed [584100] 0.163 sec
```

4.7 Processing configuration

The element `<proc>` is dedicated for the processing configuration. It comprises of the following sub-elements designed for different aspects of the processing strategy:

- `<obs>` - observations
- `<mod>` - precise models
- `<par>` - parameter configuration
- `<flt>` - estimator properties
- `<res>` - residuals checking
- `<misc>` - miscellaneous settings

Parameters configuration

The Tefnut enables estimating following parameters: receiver coordinates, receiver clock correction, zenith total delay, horizontal tropospheric gradients (in north and east directions), initial phase ambiguities, and inter-system/frequency biases (in case of multi-GNSS/multi-frequency). Whereas some parameters are estimated implicitly when becoming indispensable, such as ambiguities or inter-system/frequency biases, the estimation of other parameters can be explicitly controlled by a user. The element `<par>` thus includes various attributes for parameter specifications.

A priori standard deviation for each estimated parameter can be introduced. Such a priori sigma should optimally reflect an actual accuracy of a value used for the parameter initialization. Small sigmas with accurate a priori values can reduce the solution convergence interval. On the other hand, if the a priori values are inaccurate without a corresponding sigma specification, the Kalman filter can be confused resulting potentially even in a longer convergence than usual. The following attributes can be used for defining initial accuracy of a priori parameters, all expressed in meters:

- ***crd_sigma*** - coordinates (default option 100 m)
- ***ztd_sigma*** - zenith wet delay (default option 0.1 m)
- ***grd_sigma*** - tropospheric horizontal gradients (default option 0.0005 m)
- ***isb_sigma*** - inter-system bias (default option 1000 m)
- ***amb_sigma*** - initial phase ambiguity (recommended being left as default 1000 m)

Remaining attributes define which parameters are actually estimated:

- ***coor*** - receiver coordinates (default option EST)
- ***trop*** - tropospheric parameters (default option ZTD)
- ***iono*** - ionospheric parameters (default option NONE; currently the only option)
- ***relk*** - receiver clock corrections (default option EST; currently the only option)

The attribute ***coor*** specifies how receiver coordinates are handled in the processing. They can be estimated along with other parameters (static or kinematic), or introduced from a configuration or other input files. Currently supported options are:

= **EST** - receiver coordinates are estimated along other parameters

- = **FIX** - coordinates are fixed to values defined in configuration or other input files
- = **KIN** - coordinates are re-initialized every epoch. Such option is recommended only for highly kinematic solutions or kinematic solution with long sampling rate.

The attribute *trop* identifies which tropospheric parameters are estimated. The following enumerations represent possible options:

- = **NONE** - tropospheric delay is fully neglected (neither modeled, nor estimated),
- = **ZTD** - zenith wet delay is estimated (reported in a product output),
- = **ZTD+GRD** - zenith wet delay together with horizontal tropospheric gradients are estimated (reported in a product output),
- = **ZTD+GRD+SLT** - zenith wet delay and horizontal gradients are estimated, slant total delay are retrieved exploiting post-fit phase residuals (all reported in a product output),
- = **BLIND** - enforce introducing model tropospheric delays (using a blind model).

A configuration example specifying the estimated parameters is given below:

```
<proc>
  <par>
    coor="EST"
    rclk="EST"
    trop="ZTD+GRD+SLT"
    iono="NONE"
    crd_sigma="100"
    ztd_sigma="0.1"
    grd_sigma="0.01"
    ion_sigma="0.1"
  </par>
</proc>
```

Observations configuration

Configuring observation types is enabled in the element `<obs>` with following attributes:

- *type* - type of observations (default option COD+PHA)
- *comb* - observations combination (IONO_FREE)
- *elev* - elevation cut off (default option 10 degrees)
- *wght* - observation weighting (default option SINEL2)

Type of observations used in processing is defined in the attribute *type* with the options:

- = **COD** - only code observations
- = **PHA** - only carrier-phase observations
- = **COD+PHA** - both code and carrier-phase observations

The attribute *comb* defines a linear combination of measurements used in the processing. A single option [**IONO_FREE**] is currently enabled in Tefnut v3 when supporting the ionosphere-free linear combination only. However, new strategies for uncombined raw and multi-frequency observations will be supported in Tefnut v4.

The attribute *elev* defines an elevation angle cut off [*deg*] which is used in the processing while all observations are still decoded and kept in the software memory.

The attribute *wght* specifies strategy for observations weighting with respect to elevation angle:

- = **EQUAL** $w_{C/L} = 1$
- = **SINEL** $w_{C/L} = \frac{\sin(ele)}{\sigma_{C/L}^2}$
- = **SINEL2** $w_{C/L} = \frac{\sin(ele)^2}{\sigma_{C/L}^2}$
- = **SINEL4** $w_C = \frac{\sin(ele)^4}{\sigma_{C/L}^2}$
- = **CODPHA** $w_C = \frac{\sin(ele)^4}{\sigma_C^2}$ $w_L = \frac{\sin(ele)^2}{\sigma_L^2}$
- = **MLTPTH** - experimental function considering code multipath estimates.

where w is a diagonal element of a weight matrix, ele is a satellite elevation in degrees, and C and L stands for code and carrier-phase observation, respectively.

An example of observations configuration is given bellow:

```
<proc>
<obs
  type="COD+PHA"
  comb="IONO_FREE"
  elev="7"
  wght="SINEL2"
/>
</proc>
```

Residuals checking strategy

The element `<res>` is designed for checking carrier-phase post-fit residuals when supporting following attributes:

- **type** - type of residuals (default option NORM)
- **maxv** - normalized residual threshold (default option 3.0)

The attribute **type** controls the strategy of outliers detection by exploiting two types of post-fit residuals (original and normalized) utilized in the software:

- = **ORIG** - original (code and carrier-phase) observation residuals checked
- = **NORM** - normalized (code and carrier-phase) observation residuals analysed
- = **ALL** - both original and normalized residuals considered

The attribute **maxv** defines a threshold for identifying an outlier observation when using normalized residuals. A reasonable value for the threshold is between 2 and 3.

Remark: if original residuals are defined, the threshold for code and carrier-phase observations are configured within the constellation-specific element (Section 4.3).

An example of residuals configuration is given bellow:

```
<proc>
<res
  type="NORM"
  maxv="2.5"
/>
</proc>
```

Models configuration

The Precise Point Positioning requires precise models since it is a technique using undifferenced observations. The models are defined in the element `<mod>` with following attributes:

- ***amb_type*** - ambiguity type (default option **FLOAT**)
- ***ztd*** - blind model for ZTD (default option **SAASTAMOINEN**)
- ***ztd_mf*** - ZTD mapping function (default option **GMF**)
- ***grd_mf*** - tropospheric gradient mapping function (default option **TILTING**)
- ***attitude*** - attitude modelling (default option **NOMINAL**)
- ***eclipse*** - using/skipping satellites in eclipsing period (default option **SKIP**)
- ***pco_corr*** - used phase center offset (default option **PCO_ALL**)
- ***pcv_corr*** - used phase center variation (default option **PCV_ALL**)
- ***windup*** - phase wind-up (default option **USE**)
- ***tide_s*** - solid earth tide (default option **IERS2010**)
- ***tide_o*** - ocean tide loading (default option **IERS2010**)
- ***tide_p*** - pole tide (default option **IERS2010**)
- ***tide_a*** - atmospheric tide loading (reserved for future)

A single option in the attribute ***amb_type*** is currently **FLOAT** which means an ambiguity-float solution is only supported by Tefnut v3. Additional options are reserved for Tefnut v4.

The attribute ***ztd*** defines which blind model is used to derive a priori values for Zenith Hydrostatic Delay (ZHD) and, alternatively, for Zenith Wet Delay (ZWD):

- = **SAASTAMOINEN** - Saastamoinen model ([10]) using the GPT model ([11])
- = **HOPFIELD** - Hopfield model ([12]) with meteorological parameters from GPT model
- = **DAVIS** - Davis model ([13]) with meteorological parameters from GPT model
- = **MOPS** - Mops model summarized in [14]
- = **NONE** - ZHD as well as ZWD are set as zero value.

The attribute ***ztd_mf*** specifies a mapping function used to project slant (line-of-sight) hydrostatic and wet tropospheric delays into the corresponding parameters in zenith:

- = **COSZ** - $mf = \frac{1}{\sin(ele)}$
- = **GMF** - Global mapping function ([15])

Similarly, the attribute ***grd_mf*** specifies a mapping function for projecting slant (line-of-sight) tropospheric path delays into the linear horizontal gradient:

- = **CHEN_HERRING** - gradient mapping function developed by Chen and Herring ([16])
- = **BAR_SEVER** - gradient mapping function developed by Bar-Sever et al. ([17])
- = **TILTING** - the model based on ‘tilted’ troposphere due to azimuth-dependent asymmetry

The attribute ***attitude*** defines the satellite attitude when rotating along its axis pointing to the earth. Such a rotation is called satellite attitude orientation and depends either on satellite axes directions or external data:

- = **QUATERNIONS** - attitude angles are used from the ORBEX file in terms of quaternions.

- = **NOMINAL** - the nominal model defines satellite axes as follows: the Z axis pointing to the earth, the Y axis is along satellites solar panels, and the X axis completes the right hand oriented system. If a satellite is in eclipsing period, different attitude models for different constellations and satellite blocks are implemented.
- = **ZERO** - attitude angle is zero at every epoch. In such a case, an effect of the rotation is either neglected or assumed to be assimilated into the used precise products, such as phase biases. This model should be applied only if a user knows the same model has been applied on the precise products provider as well.

The attribute *eclipse* configures usage of satellites during eclipsing periods:

- = **USE** - affected satellites are included
- = **SKIP** - affected satellites are skipped

Remark: eclipsing period is activated if both a satellite beta angle and an orbit angle are lower than 13.5 degrees. In addition, there is a post-shadow interval defined as 1800 seconds.

The attribute *windup* activates or deactivates the utilization of the phase wind-up model:

- = **USE** - phase wind-up is applied
- = **NONE** - phase wind-up is not applied

The attributes *pco_corr* and *pcv_corr* set phase center offsets (PCO) and phase center variations (PCV), respectively:

- = **NONE** - PCO/PCV not applied
- = **RECEIVER** - only receiver PCO/PCV are applied
- = **TRANSMITTER** - only satellite PCO/PCV are applied
- = **ALL** - both receiver and satellite PCO/PCV are applied

The attributes *tide_s*, *tide_o*, and *tide_p* specifies models for solid earth tides, for ocean tide loading, and for pole tides, respectively. The models are implemented in compliance with different IERS standards which can be selected as follows:

- = **NONE** - not applied
- = **IERS96** - IERS 1996 conventions applied ([18])
- = **IERS2010** - IERS 2010 conventions applied ([19])

An example of common configuration of models is given bellow:

```
<proc>
<mod
  ztd="SAASTAMOINEN"
  ztd_mf="GMF"
  grd_mf="CHEN_HERRING"
  eclipse="SKIP"
  attitude="NOMINAL"
  windup="USE"
  tide_s="IERS2010"
  tide_o="IERS2010"
  tide_p="IERS2010"
/>
</proc>
```

Estimator properties

The Kalman filter is implemented as the main procedure for a state vector estimates. The element `<flt>` is intended to modify the Kalman filter properties and to characterize stochastic processes employed for different model parameters. The following attributes are supported:

- ***flt_method*** - an algorithm for the Kalman forward filter (default option SRCF)
- ***smt_method*** - an algorithm for the Backward smoother (default option NONE)
- ***clk_white*** - white noise of receiver clock (default option 1000 m)
- ***crd_white*** - white noise of coordinates (default option 100 m)
- ***ztd_rwalk*** - random walk of ZTD (default option $3 \text{ mm}/\sqrt{\text{hour}}$)
- ***grd_rwalk*** - random walk of trop. grad. (default option $0.3 \text{ mm}/\sqrt{\text{hour}}$)
- ***isb_rwalk*** - random walk of ISB (default option $20 \text{ mm}/\sqrt{\text{hour}}$)
- ***glo_rwalk*** - random walk of GLO ISB (default option $20 \text{ mm}/\sqrt{\text{hour}}$)
- ***gal_rwalk*** - random walk of GAL ISB (default option $20 \text{ mm}/\sqrt{\text{hour}}$)
- ***bds_rwalk*** - random walk of BDS ISB (default option $20 \text{ mm}/\sqrt{\text{hour}}$)
- ***qzs_rwalk*** - random walk of QZS ISB (default option $20 \text{ mm}/\sqrt{\text{hour}}$)
- ***amb_reset*** - time interval for ambiguities re-initialization (default option 0 s)
- ***crd_reset*** - time interval for coordinates re-initialization (default option 0 s)
- ***ztd_reset*** - time interval for ZTD re-initialization (default option 0 s)
- ***grd_reset*** - time interval for trop. gradient re-initialization (default option 0 s)
- ***par_reset*** - time interval for all parameters re-initialization (default option 0 s)

The attribute ***flt_method*** specifies an algorithm for the Kalman filter:

- = **KALMAN** - traditional Kalman filter, which suffers from round-off error. Therefore, a positive semidefinite covariance matrix is not guaranteed.
- = **SRCF** - square root covariance filter (SRCF) algorithm is numerically more stable as it uses matrix QR decomposition. As a result, round-off errors are avoided and the state vector covariance matrix is always positive semi-definite.

Additionally, the state vector at every epoch can be recalculated with the backward smoothing algorithm ([2]). While only observations from past epochs contribute to the state vector estimation in case of using the Kalman forward filter, both past and following epochs are exploited when the backward smoothing is utilized. Theoretically, such smoothing gives an alternative solution for the least squares algorithm; however, still with high temporal resolution. The smoother is initialized with the state vector at the last epoch from the previously run forward filter and recalculates all parameters sequentially in backward direction. As a result, the final estimates do not suffer from any initial convergence and formal error of all parameters are unique through the whole processing interval. The two algorithms currently implemented in the software can be selected with the attribute ***smt_method*** :

- = **RTS** - Rauch-Tung-Striebel (RTS) algorithm, [20].
- = **RTSSVD** - improved RTS algorithm utilizing matrix SVD decomposition, [2]. The original RTS algorithm suffer from round off error similarly like the Kalman filter. The improved approach uses SVD decomposition for preventing negative values on diagonal of covariance matrix.
- = **NONE** - backward smoothing is not applied.

Inter-system-bias (ISB) random walk process noise can be set in the attribute *isb_rwalk* either equally or individually for all the constellations, but GPS.

Individual parameters, or the whole state vector, can be regularly re-initialized. Such options are useful for testing or study purpose when initial convergence is enforced regularly.

Miscellaneous settings

The element `<misc>` includes attributes for miscellaneous settings, currently focused on product filtering due to either solution convergence or large formal error:

- *ztd_maxs* - maximum ZTD formal error (default option 0 m)
- *crd_maxs* - maximum coordinates formal error (default option 0 m)
- *xcl_conv* - initial convergence length for excluding (default option 0 s)

Traditional PPP float solution suffers from initial convergence for obtaining accurate estimates of the state vector. With help of this settings, the parameters that have not been estimated with an expected accuracy can be filtered out before encoding output files. A combination of three attributes *ztd_maxs* *crd_maxs* *xcl_conv* can be used for such purpose. Whereas the first two ones define the maximal formal error of zenith total delay and coordinates, the third mentioned attribute defines the convergence length explicitly.

4.8 Site metadata configuration

Metadata for individual stations can be obtained either from the SINEX [21] or from a user XML configuration file. The latter enables specifications of the following metadata types:

- *id* (string) - unique site identifier (internal software identifier)
- *name* (string) - site name (4-char/9-char station name)
- *domes* (string) - site DOMES number
- *desc* (string) - site description
- *rec* (string) - site receiver name
- *ant* (string) - site antenna+radome name
- *x* | *y* | *z* (double) - site cartesian coordinates [m]
- *dn* | *de* | *du* (double) - site North, East, and Up eccentricities [m]
- *sx* | *sy* | *sz* (double) - site coordinates sigmas [m]

Selected metadata types (*rec*, *ant*, *x/y/z*, *dn/de/du*, *sx/sy/sz*) might be attributed with a time validity when using a time-dependent subelement.

Common (or static) metadata can be set in the `<site>` parent element, i.e. without a time-dependent sub-element:

```
<site id="GOPE" name="GOPE00CZE" domes="11502M002" desc="Geodetic_Observatory_Pecny"
      x="3979316.1" y="1050312.6" z="4857067.4" dn="0.0" de="0.0" du="0.1114"
      rec="TPS_NETG3"
      ant="TPSCR.G3UUUUUUUU_TPSH"
/>
```


*Remark: the **id** is used for an internal purpose, hence it should represent a unique station identifier that can be specified in the `<gen><rec>` element or used within the `(rec)` site variable. On the other hand, the **name** represents an external station name that is used for the input and output purposes.*

As long as any individual site metadata should be valid for a particular time interval, it should be configured using time-dependent subelements (`<set>`) with a possibility to specify the start and the end timestamp explicitly in the **beg** and the **end** attribute, respectively, e.g.:

```
<site id="GOPE00CZE" name="GOPE00CZE" domes="11502M002" desc="Pecny, Ondrejov/Czech_Republic" x="3979316.1" y="1050312.5" z="4857067.1" >
  <set beg="2019-04-10T00:00:00" end="2019-04-10T00:00:00" rec="TRIMBLE_ALL0Y" ant="TPSCR.G3_0000000000TPSH" du="0.1114" dn="0.0000" de="0.0000" />
  <set beg="2018-08-08T00:00:00" end="2019-04-10T00:00:00" rec="TRIMBLE_SPS855" ant="TPSCR.G3_0000000000TPSH" du="0.1114" dn="0.0000" de="0.0000" />
  <set beg="2009-12-14T00:00:00" end="2018-08-08T00:00:00" rec="TPS_NETG3" ant="TPSCR.G3_0000000000TPSH" du="0.1114" dn="0.0000" de="0.0000" />
  <set beg="2006-07-14T00:00:00" end="2009-12-14T00:00:00" rec="ASHTECH_Z18" ant="TPSCR3_GGD_0000000000CONE" du="0.1114" dn="0.0000" de="0.0000" />
  <set beg="2006-01-24T00:00:00" end="2006-07-14T00:00:00" rec="ASHTECH_Z18" ant="ASH701946.3_0000000000SNOW" du="0.0464" dn="0.0000" de="0.0000" />
  <set beg="2000-10-04T00:00:00" end="2006-01-24T00:00:00" rec="ASHTECH_Z18" ant="ASH701946.3_0000000000SNOW" du="0.0464" dn="0.0000" de="0.0000" />
  <set beg="2000-07-24T00:00:00" end="2000-10-04T00:00:00" rec="TRIMBLE_4000SSE" ant="TRM14532.00_0000000000NONE" du="0.0464" dn="0.0000" de="0.0000" />
  <set beg="1999-11-04T00:00:00" end="2000-07-24T00:00:00" rec="ASHTECH_Z18" ant="ASH701073.3_0000000000SNOW" du="0.0464" dn="0.0000" de="0.0000" />
  <set beg="1997-06-20T00:00:00" end="1999-11-04T00:00:00" rec="TRIMBLE_4000SSE" ant="TRM14532.00_0000000000NONE" du="0.0464" dn="0.0000" de="0.0000" />
  <set beg="1997-06-11T00:00:00" end="1997-06-20T00:00:00" rec="SPP_GEOTRACER100" ant="TRM14532.00_0000000000NONE" du="0.0464" dn="0.0000" de="0.0000" />
  <set beg="1995-05-13T00:00:00" end="1997-06-11T00:00:00" rec="TRIMBLE_4000SSE" ant="TRM14532.00_0000000000NONE" du="0.0464" dn="0.0000" de="0.0000" />
</site>
```

Time-specific metadata settings should facilitate modifications and validations of station metadata in the RINEX file header. Standard IGS site-log (or GeodesyML) metadata files can be thus converted to the G-Nut settings and used via a pre-defined 'static' configuration XML file completed with a 'dynamic' input/output configuration when using the command-line arguments.

Remark: the GeodesyML site metadata file is foreseen to be supported in Tefnut v4.

5. Processing modes

The software can be run in post-processing or real-time mode. The former is default, the latter is activated with the attribute *run_rttime* of the element `<gen>` (Section 3.2).

In any case, estimated tropospheric parameters can be encoded into SINEX_TRO 2.0 product files [22] defined by the element `<out><trosinex>`.

5.1 Post-processing mode

A post-processing solution is based on files only. As the PPP works on undifferenced observations from a single (or more) receiver(s), i.e. autonomously, precise satellite orbits and clocks are mandatory input products. Observation data and precise products of satellites orbits and clock corrections can be read from one of these combinations of input formats:

- `<inp><sp3>` - satellite orbits and low-rate clocks in *SP3* data file
- `<inp><rinexc>` - satellite high-rate clocks in *Clk RINEX* data file
- `<inp><rinexn>` - satellite navigation in *Nav RINEX* data file(s)
- `<inp><rinexo>` - receiver observation in *Obs RINEX* data file(s)

or

- `<inp><rinexo>` - for receiver observation in *Obs RINEX* data files
- `<inp><rinexn>` - for satellite navigation in *Nav RINEX* data files
- `<inp><bncrtcem>` - satellite orbit and clock corrections (w.r.t. navigation data) obtained from *RTCM3* streams and stored by the BNC software [5] in *BNC SSR* data files

5.2 Real-time mode

In case of real-time processing, both input data and products stems from data streams read in an infinite loop. These input formats are supported in Tefnut v3:

- `<inp><rtcm>` - receiver observation data in *RTCM3* [4] data stream(s)
- `<inp><rtcm>` - satellite navigation data in *RTCM3* [4] data stream
- `<inp><rtcm>` - satellite orbit and clock corrections in *RTCM3* [4] data stream

where several RTCM streams are individually used for different inputs (observation data streams are available site by site), or

- `<inp><rinexn>` - satellite navigation data in *Nav RINEX*-like data stream
- `<inp><bncrtcem>` - satellite orbit and clock corrections in *BNC-SSR* [5] data stream
- `<inp><bncobs>` - receiver observations in *BNC OBS* [5] data stream(s)

where different ASCII input format data streams are used (observation data might be provided synchronously in a single data stream).

Both observations as well as precise products are delayed due to data dissemination, the former in seconds, the latter in several tens of seconds. Hence, a processing delay should be effectively set up with the attribute ***run_delay*** [*sec*] of the the element `<gen>` . A reasonable value corresponds to 40-60 seconds.

Traditionally, by re-calculating solutions achieved with the forward filter, the backward smoothing algorithm is able to further enhance the results (Section 4.7). The smoother can be also activated in the real-time mode when targeting a Near Real-Time (NRT) solution. A latency of re-calculation can be specified with the attribute ***nrt_delay*** [*sec*] of the `<gen>` element. It represents an addition to the latency of real-time data described above and, implicitly, defines an interval length of the smoothing. The longer the interval is, the more accurate results can be obtained.

Since data are decoded continuously (and stored in real time), a memory storage must be cleaned regularly. For this purpose, the attribute ***max_epoch*** of the element `<gen>` defines a maximum number of data/product epochs which are kept in the memory.

Various settings can be re-configured on-the-fly, i.e. without killing the process (Section 7). This is highly beneficial for preventing a necessary re-convergence after an extra cold start.

5.3 Parallel vs. sequential processing

Multiple stations can be processed sequentially, or in parallel. The parallel processing exploits computational resources effectively by creating dedicated processing threads, and potentially using multiple processing cores (if these are available). A sequential mode is requested for PPP threads, station-wise processing, and input file reading when the attribute ***max_threads*** , ***max_sites*** , or ***max_files*** equals 1, respectively. Any value > 1 represents a user-defined parallel configuration (explicitly). The NULL value ('0') suggests an auto-selection when exploiting all available resources (CPU), number of files and stations (implicitly).

Remark: user is responsible for an effective settings considering other loads on the same server.

5.4 Handling station/receiver names

Data are processed station by station in PPP, however, observations from many stations can be stored and processed simultaneously. In such a case, active stations can be defined explicitly in the element `<gen><rec>` .

When the element is empty, the list of sitenames is generated implicitly, but differently for post-processing and real-time modes. In case of the post-processing, all stations found in corresponding internal data containers are processed, either collected from RINEX files or a user configuration. In case of real-time, only stations with metadata provided in a user configuration (Section 4.8) can be processed.

Remark: station/receiver identifier in `<gen><rec>` represents an internal string code that should be unique for linking site-specific metadata, data, products and models. It may be repre-

sented with an arbitrary string only if station metadata are defined in the XML configuration, else it should correspond to a true sitename as available e.g. from the RINEX files.

When reading a RINEX observation file in the post-processing mode, the MARKER string in a data header is considered to be a sitename. However, the MARKER may not be filled properly, or it may be inconsistent with the sitename used in the filename, e.g. 4-char in the RINEX MARKER vs. 9-char in a RINEX long filename. The Tefnut guesses a correct sitename by confronting the MARKER with the filename and the `<gen><rec>` user settings (later in sequence, a high priority). The guessed name is reported in the log file and used in the software.

In the real-time mode, the NTRIP mountpoint can also be used to define sitename as well, however, it can be overwritten explicitly by defining the sitename within the ***id*** attribute of the format specification subelement, e.g. `<inp><rtcm>` .

A user can also define sitename explicitly for any individual input file or data stream by using an attribute ***id*** thus giving preference over any information either from the RINEX MARKER, filename, or NTRIP mountpoint.

6. Output formats

The Tefnut provides several output formats which contents are described in following sections:

Log outputs – general log and site-specific PPP logs both in ASCII formats, see Section 6.1.

Solution outputs – site-specific PPP result files in the GNUT format, see Section 6.2.

Troposphere outputs – troposphere parameters in the standard format, see Section 6.3.

Additionally, two attributes are part of the `<elem>` configuration

- **verb** (boolean) - verbosity level for log files (default: 0),
- **append** (boolean) - append mode for log files (default: *false*).

Several other attributes can be used in the configuration of any format element:

- **upd** (integer) - file update (default: data product length [s]; minimum is 60s),
- **len** (integer) - data product period length (default: processing period [s]),
- **smp** (integer) - data product sampling rate (default: processing rate [s]),
- **tsys** (UTC,LOC) - time system for output file (default = *UTC*),
- **toff** (integer) - time offset for output file (default = 0s).

Any output mask may include time-/site-specific variables, see Section 3.3. This is particularly needed for the real-time operation, however, it can be used for the post-processing solution too, e.g. for splitting daily processing results into sub-daily output files.

The following outputs are generated using this sample configuration:

```
<?xml version="1.0"?>
<config>
  <gen>
    <int> 30                </int>
    <rec> GOPE00CZE        </rec>
    <beg> 2021-04-10 00:00:00 </beg>
    <end> 2021-04-10 23:59:30 </end>
    <sys> GPS GLO GAL      </sys>
  </gen>
  <inp>
    <rinexo> INP/(REC9)_R_%Y%j0000_01D_30S_M0.crx.gz </rinexo>
    <rinexn> INP/BRDC00GOP_R_%Y%j0000_01D_MN.rnx.gz </rinexn>
    <rinexc> INP/CODOMGXFIN_%Y%j0000_01D_30S_CLK.CLK.gz </rinexc>
    <sp3> INP/CODOMGXFIN_%Y%j0000_01D_05M_ORB.SP3.gz </sp3>
    <erp> INP/CODOMGXFIN_%Y%j0000_03D_12H_ERP.ERP.gz </erp>
    <sinex> INP/euref.snx </sinex>
    <atx> INP/igs14_2215.atx </atx>
    <blq> INP/EPN_FES2004.BLQ </blq>
  </inp>
  <out>
    verb="1"
    <trosinex smp="300"> OUT/tef-pp-%Y-%m-%d.tro.gz </trosinex>
    <log> OUT/tef-pp-%Y-%m-%d.log </log>
    <ppp> OUT/(REC9)-%Y-%m-%d.ppp </ppp>
    <smt> OUT/tef-pp-%Y-%m-%d.smt </smt>
    <flt> OUT/tef-pp-%Y-%m-%d.flt </flt>
    <tef> OUT/tef-pp-%Y-%m-%d.tef </tef>
  </out>
  <proc>
  <par>
```

```

trop="ZTD+GRD+SLT"
iono="NONE"
coor="EST"
crd_sigma="100"
grd_sigma="0.0005"
amb_sigma="1000"
ion_sigma="100"
ztd_sigma="0.1"
isb_sigma="1" />
<obs type="COD+PHA" wght="SINEL2" elev="7" comb="IONO_FREE" />
<res type="NORM" maxv="2" />
<mod
  ztd_mf="GMF" ztd="SAASTAMOINEN" grd_mf="CHEN_HERRING"
  ion_mf="ICOSZ"
  tide_s="IERS2010" windup="USE"
  tide_o="IERS2010" eclipse="SKIP" pco_corr="ALL"
  tide_p="IERS2010" attitude="NOMINAL" pcv_corr="ALL"
  tide_a="NONE"
/>
<flt
  flt_method="srcf" smt_method="rtssvd"
  clk_white="1000" crd_white="100" ion_white="5"
  ztd_rwalk="3" grd_rwalk="0.3" isb_rwalk="20"
  amb_reset="0" par_reset="0" crd_reset="0" ztd_reset="0" grd_reset="0" />
</proc>
<nav
  chk_record="true" chk_health="true" chk_duplic="true"
  chk_header="false" chk_params="false" chk_others="false"
  sig_bounds="2" chk_tot="true" key_tot="false"
/>
<sys id="GPS" sigma_c="2" sigma_l="0.01" slip_thr="2" />
<sys id="GAL" sigma_c="1" sigma_l="0.01" slip_thr="2" />
<sys id="GLO" sigma_c="10" sigma_l="0.02" slip_thr="2" />
<sys id="BDS" sigma_c="2" sigma_l="0.02" slip_thr="2" />
<site
  id="GOPE00CZE"
  name="GOPE00CZE"
  domes="11502M002"
  desc="Pecny, Ondrejov, CZ/Czechia">
  <set
    beg="2019-04-10T00:00:00"
    rec="TRIMBLE_ALLOY"
    ant="TPSCR.G3_TPSH"
    x="3979316.100"
    y="1050312.500"
    z="4857067.100"
    du="0.1114"
    dn="0.0000"
    de="0.0000" />
  <set
    beg="2018-08-08T00:00:00"
    end="2019-04-10T00:00:00"
    rec="TRIMBLE_SPS855"
    ant="TPSCR.G3_TPSH"
    x="3979316.100"
    y="1050312.500"
    z="4857067.100"
    du="0.1114"
    dn="0.0000"
    de="0.0000" />
  </site>
</config>

```

6.1 Log outputs

There are two log levels - a generic one in a single file and PPP-process ones for each individual stations.

The former reports all generic messages, warnings, and errors.

```

2022-09-11 22:38:10 [gsetbase:0] program: G-Nut/Tefnut RT [3.3] Sep 8 2022 12:44:27 (Rev: 3769:3770M)
2022-09-11 22:38:10 [gsetbase:0] =====
2022-09-11 22:38:10 [gsetbase:1] setting: tef-pp-2021-04-10.cfg.SLT
2022-09-11 22:38:10 [gsetbase:1] logfile: OUT/tef-pp-%Y-%m-%d.log
2022-09-11 22:38:10 [gsetbase:1] verbose: 1
2022-09-11 22:38:10 [gsetbase:1] =====
2022-09-11 22:38:10 [gsetrec:1] META crd GOPE00CZE from 2018-08-08 00:00:00 to next | 3979316.1000 1050312.5000 4857067.1000 | -999 -999 -999
2022-09-11 22:38:10 [gsetrec:1] META crd GOPE00CZE from 2019-04-10 00:00:00 to next | 3979316.1000 1050312.5000 4857067.1000 | -999 -999 -999
2022-09-11 22:38:10 [gsetrec:1] META ecc GOPE00CZE from 2018-08-08 00:00:00 to next | 0.0000 0.0000 0.1114
2022-09-11 22:38:10 [gsetrec:1] META ecc GOPE00CZE from 2019-04-10 00:00:00 to next | 0.0000 0.0000 0.1114
2022-09-11 22:38:10 [gsetrec:1] META ant GOPE00CZE from 1980-01-01 00:00:00 to next | undef
2022-09-11 22:38:10 [gsetrec:1] META ant GOPE00CZE from 2018-08-08 00:00:00 to next | TPSCR.G3 TPSH
2022-09-11 22:38:10 [gsetrec:1] META ant GOPE00CZE from 2019-04-10 00:00:00 to next | TPSCR.G3 TPSH
2022-09-11 22:38:10 [gsetrec:1] META rec GOPE00CZE from 1980-01-01 00:00:00 to next | undef
2022-09-11 22:38:10 [gsetrec:1] META rec GOPE00CZE from 2018-08-08 00:00:00 to next | TRIMBLE SPS855
2022-09-11 22:38:10 [gsetrec:1] META rec GOPE00CZE from 2019-04-10 00:00:00 to next | TRIMBLE ALLOY
2022-09-11 22:38:10 [main:1] creating object: GOPE00CZE
2022-09-11 22:38:27 [gserv_ios:0] READ ATX [-]: INP/igs14_2215.atx completed [22457040] 16.912 sec
2022-09-11 22:38:27 [gserv_ios:0] READ BLQ [-]: INP/EPN_FES2004.BLQ completed [322042] 0.019 sec
2022-09-11 22:38:27 [sinex:1] USER requested long sitename: GOPE00CZE <- GOPE
2022-09-11 22:38:27 [gserv_ios:0] READ SINEX [-]: INP/euref.snx completed [670732] 0.136 sec
2022-09-11 22:38:27 [sinex:0] Error: SINEX:ECC !! GOPE00CZE - object conflict 2006-01-24 11:00:00 ECC: 0.0000 0.0000 0.0464 !! 0.0000 0.0000 0.0467
2022-09-11 22:38:27 [sinex:0] Error: SINEX:ECC !! GOPE00CZE - object conflict 2006-07-14 12:50:00 ECC: 0.0000 0.0000 0.0464 !! 0.0000 0.0000 0.1114
2022-09-11 22:38:27 [sinex:0] Error: SINEX:ECC !! GOPE00CZE - object conflict 2009-12-14 10:30:00 ECC: 0.0000 0.0000 0.0464 !! 0.0000 0.0000 0.1114
2022-09-11 22:38:27 [sinex:0] Error: SINEX:ECC !! GOPE00CZE - object conflict 2018-07-17 19:08:00 ECC: 0.0000 0.0000 0.0464 !! 0.0000 0.0000 0.1114
2022-09-11 22:38:28 [gserv_ios:0] READ RINEXN [-]: INP/BRDC00GDP_R_20211000000_01D_MN.rnx.gz completed [5997424] 0.804 sec
2022-09-11 22:38:31 [gserv_ios:0] READ RINEXC [-]: INP/CODOMGXFIN_20211000000_01D_30S_CLK.CLK.gz completed [35663940] 3.492 sec
2022-09-11 22:38:31 [sp3:0] READ sp3: beg:2021-04-10 00:00:00 epo:289 acr:AIUB sys:IGb14 flg:P dat:d+d orb:FIT
2022-09-11 22:38:31 [gserv_ios:0] READ SP3 [-]: INP/CODOMGXFIN_20211000000_01D_05M_ORB.SP3.gz completed [2055963] 0.373 sec
2022-09-11 22:38:31 [gserv_ios:0] READ ERP [-]: INP/CODOMGXFIN_20211000000_03D_12H_ERP.ERP.gz completed [1630] 0.000 sec
2022-09-11 22:38:31 [rinexo:1] FILE long site name: GOPE00CZE (9-CH)
2022-09-11 22:38:34 [gserv_ios:0] READ RINEXO [-]: INP/GOPE00CZE_R_20211000000_01D_30S_MO.crx.gz completed [23463221] 2.525 sec
2022-09-11 22:38:34 [rinexo:0] Message: GLO BIASES unknown .. file://GOPE00CZE_R_20211000000_01D_30S_MO.crx.gz
2022-09-11 22:38:34 [main:0] PP filter started
2022-09-11 22:40:30 [main:0] PP filter finished 116.000 sec
2022-09-11 22:40:30 [main:0] total time: 140.703 / 140.508 sec
2022-09-11 22:40:30 [gserv_ios:1] USER TEF output update [86400]
2022-09-11 22:40:31 [gserv_ios:0] SAVE TEF [-]: OUT/tef-pp-2021-04-10.tef completed [955724] 0.556 sec
2022-09-11 22:40:31 [gserv_ios:1] USER SMT output update [86400]
2022-09-11 22:40:31 [gserv_ios:0] SAVE SMT [-]: OUT/tef-pp-2021-04-10.smt completed [477812] 0.109 sec
2022-09-11 22:40:31 [gserv_ios:1] USER TROSINEX output update [86400]
2022-09-11 22:40:31 [gserv_ios:0] SAVE TROSINEX [-]: OUT/tef-pp-2021-04-10.tro.gz completed [914137] 0.207 sec
2022-09-11 22:40:31 [gserv_ios:1] USER FLT output update [86400]
2022-09-11 22:40:31 [gserv_ios:0] SAVE FLT [-]: OUT/tef-pp-2021-04-10.flt completed [478144] 0.091 sec

```

The log verbosity has been set to 1, while the verbosity is reflected in initial bracket along with the object class identifier reporting it. After the initial setting information, station meta data available from the G-Nut XML configuration are reported for the object GOPE00CZE. Then a sequence of all input files were read successfully accompanied with a file size and a reading time. From the long file name as well as from the user settings, the 9-char station name (GOPE00CZE) could have been recognized. The object conflicts are due to reading SINEX meta data in addition to the G-Nut XML configuration which was incomplete (only the last two records were kept). When reading RINEXO file, the message informs that the 'GLO BIASES' field in the header was not available. Then the PPP filter reported start and end of the process as well as the total length. Finally, outputs were written in configured files (using the specific output attribute settings).

The PPP log files summarize outputs from individual PPP processes of each station.

```

[gsppflt:1] GOPE00CZE filter: (SRCF) 2021-04-10 00:00:00
[gsppflt:1] GOPE00CZE epoch 2021-04-10 00:00:00 was not calculated
[gtide2010:0] WARNING: Site GOPE00CZE not found in ocean tide BLQ file!
[gsppflt:1] GOPE00CZE processing epoch: 2021-04-10 00:00:30.000
[gsppflt:1] GOPE00CZE processing epoch: 2021-04-10 00:01:00.000
[gsppflt:1] GOPE00CZE processing epoch: 2021-04-10 00:01:30.000
[gsppflt:1] GOPE00CZE processing epoch: 2021-04-10 00:02:00.000
[gsppflt:1] GOPE00CZE processing epoch: 2021-04-10 00:02:30.000
[gsppflt:1] GOPE00CZE processing epoch: 2021-04-10 00:03:00.000
...
[gsppflt:1] GOPE00CZE processing epoch: 2021-04-10 00:09:00.000
[gspp:1] GOPE00CZE outlier (Norm residual: phase) E27 size:26 v: 38.665 ele: 7.73 00:09:30.000
[gsppflt:1] GOPE00CZE processing epoch: 2021-04-10 00:09:30.000
[gsppflt:1] GOPE00CZE processing epoch: 2021-04-10 00:10:00.000
[gsppflt:1] GOPE00CZE processing epoch: 2021-04-10 00:10:30.000
[gsppflt:1] GOPE00CZE processing epoch: 2021-04-10 00:11:00.000
[gspp:1] GOPE00CZE outlier (Norm residual: phase) E27 size:27 v: 206.032 ele: 7.54 00:11:30.000
[gsppflt:1] GOPE00CZE processing epoch: 2021-04-10 00:11:30.000
[gsppflt:1] GOPE00CZE processing epoch: 2021-04-10 00:12:00.000
...
[gsppflt:1] GOPE00CZE processing epoch: 2021-04-10 23:59:00.000
[gsppflt:1] GOPE00CZE processing epoch: 2021-04-10 23:59:30.000
[gsppsm:1] GOPE00CZE smoother: (RTSSVD) 2021-04-10 23:59:30

```

6.2 Solution outputs

The PPP solution outputs are provided in site-specific files, hence the site-expansion should be used in the output file path. Thus variable (rec) or (REC) should be part of the file name, see Section 3.3.

The solution file consists of the header and the data parts. The header includes important information such as

- =**VER** - format version
- =**GEN** - generation timestamp
- =**PGM** - program version
- =**EPO** - epoch of reference coordinates
- =**SIT** - site name + domes
- =**SYS** - reference system (obtained from precise products)
- =**XYZ** - reference coordinates in XYZ (m)
- =**BLH** - reference coordinates in BLH (m)

There are three possible data contents of the solution output format, which can be distinguished by three format (element) identifiers:

- <flt> - results of the forward filter
- <smt> - results of the backward smoothing
- <tef> - results from both forward filter and backward smoothing

It is possible to set one of the solution output files or all of them simultaneously and, in real-time operation, with even different update rate or sampling.

Data lines are distinguished by the prefix identifier too:

- =**FLT** - results of the forward filter
- =**SMT** - results of the backward smoother

The data lines starting with the =**FLT** and =**SMT** prefix contains the same format which includes:

- **YYYY-MM-DD HR:MI:SC** - data epoch
- **dX, dY, dZ** - coordinate residuals in XYZ [m] (w.r.t. the reference coordinates)
- **ZTD** - zenith total delay [m]
- **GN, GE** - tropospheric horizontal (linear) gradients in North and East direction [mm]
- **CLK** - receiver clock estimates [m]
- **sX, sY, sZ** - formal error of coordinate estimates in XYZ [m]
- **sZTD** - formal error of zenith total delay [m]
- **#sat, #ex** - number of all used satellites in the solution and excluded satellites
- **GDOP** - number of all and excluded satellites
- **Sig0** - a posteriori unique sigma

- **AR** - ambiguity resolution indicator (0=float, >0 number of fixed satellites)
- **Ratio** - ratio of ambiguity-fixed first vs. second best solution

Samples of both <flt> and <smt> files from a unique solution are provided below. A convergence can be clearly identified in the former, while eliminated by a backward smoothing in the latter (unfortunately still not completed perfectly to the first epoch).

```
=VER 1.00
=GEN 2022-09-11 11:58:39
=PGM G-Nut/Tefnut RT [3.3]
=EPO 2021-04-10 12:00:00
=SIT GQPE00CZE 11502M002
=SYS Igb14
=XYZ 3979315.8743 1050312.7502 4857067.2692
=BLH 49.913706844 14.785626927 592.6118

=HDR YYYY-MM-DD HR:MI:SC dX[m] dY[m] dZ[m] ZTD[m] GN[mm] GE[mm] CLK[m] sX[m] sY[m] sZ[m] sZTD[m] #sat #ew GDOP SigO AR Ratio
=FLT 2021-04-10 00:00:30 0.4223 0.1231 0.3471 2.1943 0.0006 0.0007 130413.500 1.76284 1.25098 2.09554 0.09974 27 2 1.0 0.3 0 0.00
=FLT 2021-04-10 00:01:00 0.3132 0.0665 0.4066 2.2041 0.0016 0.0020 130412.915 1.22060 0.87555 1.32358 0.09717 27 2 1.0 0.3 0 0.00
=FLT 2021-04-10 00:01:30 0.1328 -0.0625 0.2653 2.2075 0.0014 0.0027 130412.522 0.96461 0.70269 0.95335 0.09111 27 2 1.0 0.2 0 0.00
=FLT 2021-04-10 00:02:00 0.0154 0.0202 0.0950 2.2049 0.0031 0.0023 130412.238 0.80225 0.59531 0.73666 0.08224 27 2 1.0 0.2 0 0.00
=FLT 2021-04-10 00:02:30 -0.0949 -0.0656 0.0176 2.2013 0.0025 0.0039 130412.007 0.68470 0.51904 0.59557 0.07230 27 2 1.0 0.2 0 0.00
=FLT 2021-04-10 00:03:00 -0.1500 -0.0222 0.0196 2.2029 0.0025 0.0048 130411.883 0.59324 0.46059 0.49593 0.06277 27 2 1.0 0.3 0 0.00
```

```
=VER 1.00
=GEN 2022-09-11 11:58:39
=PGM G-Nut/Tefnut RT [3.3]
=EPO 2021-04-10 12:00:00
=SIT GQPE00CZE 11502M002
=SYS Igb14
=XYZ 3979315.8712 1050312.7429 4857067.2732
=BLH 49.913706900 14.785626840 592.6117

=HDR YYYY-MM-DD HR:MI:SC dX[m] dY[m] dZ[m] ZTD[m] GN[mm] GE[mm] CLK[m] sX[m] sY[m] sZ[m] sZTD[m] #sat #ew GDOP SigO AR Ratio
=SMT 2021-04-10 00:01:30 -0.0035 -0.0066 0.0099 2.2133 -0.0562 -0.1446 130411.105 0.01388 0.00853 0.01606 0.00189 28 4 1.0 0.1 0 0.00
=SMT 2021-04-10 00:02:00 -0.0075 -0.0052 0.0069 2.2134 -0.0562 -0.1464 130411.100 0.01389 0.00853 0.01605 0.00187 28 4 1.0 0.1 0 0.00
=SMT 2021-04-10 00:02:30 0.0014 -0.0077 0.0116 2.2134 -0.0565 -0.1480 130411.107 0.01391 0.00854 0.01604 0.00186 28 4 1.0 0.1 0 0.00
=SMT 2021-04-10 00:03:00 -0.0010 -0.0091 0.0132 2.2135 -0.0568 -0.1496 130411.108 0.01392 0.00854 0.01603 0.00184 28 4 1.0 0.1 0 0.00
```

6.3 Troposphere outputs

PPP troposphere estimates are optimally provided in the SINEX_TRO format which is a commonly adopted standard in IGS. Detailed format description can be found at https://files.igs.org/pub/data/format/sinex_tro_v2.00.pdf, hence only principle notes are summarized here.

The format consists of blocks from which most are headers containing meta data information and data containing estimated parameters.

The header part includes the following mandatory blocks:

- **FILE/REFERENCE** - file meta data
- **TROP/DESCRIPTION** - details on troposphere estimated models and parameters
- **SITE/ID** - station identification
- **SITE/COORDINATES** - station coordinates
- **SITE/ECCENTRICITIES** - station eccentricities
- **SITE/ANTENNA** - station antennas
- **SITE/RECEIVER** - station receivers

The data part includes a mandatory block for epoch-wise troposphere estimates (optionally other meteorological and auxiliary parameters) and an optional block for slant delay retrievals and information (if requested):

- **TROP/SOLUTION** - epoch-wise troposphere parameter estimates
- **SLANT/SOLUTION** - satellite-/epoch-wise slant delay retrievals and information

```

%=TRO 2.00 GOP 2022:254:43119 GOP 2021:100:00090 2021:100:86340 P 3 2 S
*-----*
+FILE/REFERENCE
*INFO_TYPE INFO-----
DESCRIPTION GOP - Geodetic Observatory Pecny, RIGTC
OUTPUT Solution parameters
CONTACT gnss@pecny.cz
SOFTWARE G-Nut/Tefnut RT [3.3] Sep 8 2022 12:44:27 (Rev: 3769:3770M)
INPUT GNSS data
VERSION NUMBER 001
-FILE/REFERENCE
*-----*
+TROP/DESCRIPTION
*-----*
KEYWORD VALUE(S)-----
TROPO SAMPLING INTERVAL 300
SLANT SAMPLING INTERVAL 300
DATA SAMPLING INTERVAL 30
GNSS SYSTEMS ERG
TROPO MODELING METHOD KALMAN FILTER/SMOOTHER
GEOID MODEL GPT/EGM96
OCEAN TIDE LOADING MODEL FES2004
ATMOSPH TIDE LOADING MODEL NOT APPLIED
ELEVATION CUTOFF ANGLE 7.000
OBSERVATION WEIGHTING SINEL2
A PRIORI TROPOSPHERE SAASTAMOINEN
TROPO MAPPING FUNCTION GMF
GRADS MAPPING FUNCTION CHEN_HERRING
TROPO PARAMETER NAMES TROTOT STDDEV TRODRY TROWET TGNTOT STDDEV TGETOT STDDEV NSAT GDOP
TROPO PARAMETER UNITS 1e+03 1e+03 1e+03 1e+03 1e+03 1e+03 1e+03 1e+03 1 1
TROPO PARAMETER WIDTH 6 6 6 6 6 6 6 6 4 4
SLANT PARAMETER NAMES SLTTOT STDDEV SLTDRY SLTWET SLTIWV SLTGRD SATRES SATMPT SAT SATELE SATAZI FACDRY FACWET FACGRD
SLANT PARAMETER UNITS 1e+03 1e+03 1e+03 1e+03 1 1e+03 1e+03 1e+03 1 1 1 1 1 1
SLANT PARAMETER WIDTH 8 6 8 6 6 6 6 6 4 7 7 9 9 8
-TROP/DESCRIPTION
*-----*
+SITE/ID
*STATION__ PT __DOMES__ T _STATION DESCRIPTION__ _LONGITUDE _LATITUDE _HGT_ELI_ _HGT_MSL_
GOPEOOCZE A 11502M002 P Pecny, Ondrejov / CZ/C 14.785627 49.913707 592.730 630.516
-SITE/ID
*-----*
+SITE/COORDINATES
*STATION__ PT SOLN T __DATA_START__ __DATA_END__ __STA_X__ __STA_Y__ __STA_Z__ SYSTEM REMRK
GOPEOOCZE A 1 P 2021:100:00090 2021:100:86340 3979315.882 1050312.761 4857067.269 IGB14 GOP
-SITE/COORDINATES
*-----*
+SITE/ECCENTRICITY
*STATION__ PT SOLN T DATA_START__ DATA_END__ AXE MARKER->ARP(M)
GOPEOOCZE A 1 P 2021:100:00090 2021:100:86340 UME 0.1114 0.0000 0.0000
-SITE/ECCENTRICITY
*-----*
+SITE/ANTENNA
*STATION__ PT SOLN T __DATA_START__ __DATA_END__ DESCRIPTION__ S/N__
GOPEOOCZE A 1 P 2021:100:00090 2021:100:86340 TPSCR.G3 TPSH ----- IGS14_2215
-SITE/ANTENNA
*-----*
+SITE/RECEIVER
*STATION__ PT SOLN T __DATA_START__ __DATA_END__ DESCRIPTION__ S/N__ FIRMW__
GOPEOOCZE A 1 P 2021:100:00090 2021:100:86340 TRIMBLE ALLOY -----
-SITE/RECEIVER
*-----*
+TROP/SOLUTION
*STATION__ EPOCH__ TROTOT STDDEV TRODRY TROWET TGNTOT STDDEV TGETOT STDDEV NSAT GDOP
GOPEOOCZE 2021:100:00300 2213.5 1.8 2162.9 50.6 -0.16 0.21 -0.06 0.20 25 1.0
GOPEOOCZE 2021:100:00600 2213.7 1.7 2162.9 50.7 -0.19 0.20 -0.06 0.19 26 1.0
GOPEOOCZE 2021:100:00900 2213.9 1.6 2162.9 51.0 -0.23 0.20 -0.07 0.18 27 0.9
GOPEOOCZE 2021:100:01200 2214.0 1.5 2162.9 51.1 -0.25 0.19 -0.08 0.17 25 1.0
GOPEOOCZE 2021:100:01500 2214.1 1.5 2163.0 51.1 -0.27 0.19 -0.11 0.17 24 1.1
GOPEOOCZE 2021:100:01800 2214.0 1.5 2162.9 51.1 -0.28 0.19 -0.15 0.17 23 1.1
...
-TROP/SOLUTION
*-----*
+SLANT/SOLUTION
*STATION__ EPOCH__ SLTTOT STDDEV SLTDRY SLTWET SLTIWV SLTGRD SATRES SATMPT SAT SATELE SATAZI FACDRY FACWET FACGRD
GOPEOOCZE 2021:100:00300 2225.3 1.8 2163.5 50.6 0.0 -0.0 11.3 0.0 E04 88.728 333.655 1.000246 1.000246 0.0222
GOPEOOCZE 2021:100:00300 4009.3 2.4 3901.9 91.3 0.0 0.3 15.7 0.0 E09 33.561 247.999 1.803977 1.806644 2.7031
GOPEOOCZE 2021:100:00300 2711.9 2.0 2631.5 61.5 0.0 0.1 18.8 0.0 E11 55.232 223.156 1.216622 1.217006 0.8428
GOPEOOCZE 2021:100:00300 7429.4 3.3 7231.3 170.1 0.0 1.6 26.4 0.0 E12 17.185 173.343 3.343286 3.365367 10.5731
GOPEOOCZE 2021:100:00300 4460.8 2.5 4346.5 101.8 0.0 -0.4 13.0 0.0 E19 29.723 65.280 2.009518 2.013517 3.4932
GOPEOOCZE 2021:100:00300 8177.2 3.4 7961.4 187.6 0.0 -1.4 29.6 0.0 E21 15.522 71.242 3.680827 3.710717 12.8989
GOPEOOCZE 2021:100:00300 14881.9 4.6 14516.9 348.7 0.0 -7.2 23.6 0.0 E27 8.126 21.893 6.711648 6.897562 42.7634
...
-SLANT/SOLUTION
%=ENDTRO

```

7. User console

A user console is a built-in tool intended for a remote access to control any operational processing, thus useful for real-time processing mode only. The console can be used for checking data in the software memory (internal data and product containers), for monitoring input data including decoding processes, and for checking or re-configuring user settings. Multiple users may login the console simultaneously.

A login access is generally expected from a localhost server. The console login can however request a username, password, and port number, while the port is a mandatory one. If the attributes *user* and *pasw* are set, the console prompts for a user credentials during the login. A console configuration can be then specified in the XML configuration via the element `<console>` and its corresponding attributes.

```
<console
  user="user"           # optional
  pasw="pasw"          # optional
  port="4444"          # mandatory
/>
```

The command-line arguments can be also used according to rules defined in Section 3:

```
:console:user="user"
:console:pasw="pasw"
:console:port="4444"
```

The console operations are performed with commands executed in the terminal. The tool **rlwrap** can be utilized for efficient handling of the commands history making the usage of the console significantly more comfortable. When utilizing **telnet** for the connection to a localhost (both tools need to be installed), a corresponding command can be performed as follows:

```
rlwrap telnet localhost <port>
```

Basic console commands are listed with the keyword *help*.

```
> help
-----
G-Nut console help
-----
      help           - this help
      exit           - close client
      info           - show client info and settings
      verb <int>    - select verbosity
      watch <cmd>   - periodically watch a command (see "watch_help", for exit use "watch_stop")
-----
Internal data containers
-----
      list data      - list available data containers
      set data <key> - set active container
      show <args>    - monitor elements of chosen container, start with "show_help"
-----
Internal data decoders
-----
      list coder     - list available data decoders
      set coder <key> - set active decoder
      show           - monitor data decoding, start with "show_help"
-----
Configuration (experimental)
-----
      config         - show full configuration
      config <commands> - command-line settings (e.g. config :out:verb=2 :gen:int=300)
      reload         - reload original configuration file
```

The *exit* command is called automatically after 10 min of inactivity. Some of the console outputs depend on the verbosity settings *verb* (0 by default).

The *info* command shows information about the connection, its verbosity setting, *watch* interval (Section 7.3), and data object selection (Sections 7.1 and 7.2):

```
G-Nut console connect information:
      IP address: 127.0.0.1
      Port:      4444
      Client:    8
      Verbosity: 2
      Watch int.[s]: 5
      Data object: orb
```

7.1 Data container browsing

The console allows users to browse any data or product internal structures in the process memory. All available data types and containers can be reported by the command **list data**:

```
> list data
G-Nut console data structures:
      (obs) - GRP_OBSERV | ALLOBS
      (orb) - GRP_EPHEM | ALLPREC
> set data obs
Selected container: obs
```

Each of the listed line consists of: (*<key>*) - *data type* | *container type*. The command **set data <key>** then selects the particular container to work with. A single selection is possible.

Available commands for the selected structure can be listed with the command **show help**:

```
> show help
-----
Container options
-----
show + ...
      help      - this help
      stats     - basic statistics (verb dependent)
      <key> <args> - show container structure contents (depending on verb, given key and args)
-----
Container structure keys
-----
-> sites
-> epochs (use "last" for last epoch)
-> satellites (sats)
-> observations (obs)
```

While the first part (*Container options*) lists all generic commands, the second part (*Container structure keys*) lists only the keys available for the selected structure. Such keys correspond to the data and its internal hierarchical representation; therefore, when requesting a particular key, all the preceding have to be already specified too (see examples below).

Hence, the list of stations available in the structure is printed with the command **show sites**:

```
> show sites
Number of sites: 46
      MATE00ITAO ( epochs: 12)
      ZIM200CHEO ( epochs: 19)
      WTZR00DEUO ( epochs: 15)
```

The epochs of the selected station MATE00ITA0 can be browsed with the extended command:

```
> show epochs MATE00ITA0
MATE00ITA0
    2021-11-29 10:48:15
    2021-11-29 10:48:20
    2021-11-29 10:48:25
    2021-11-29 10:48:30
```

Printing observations of a particular epoch for a station and satellite can be then obtained:

```
> show obs MATE00ITA0 "2021-11-29_11:05:20" G02
MATE00ITA0 2021-11-29 11:05:20 G02
    C1C 23503970.185938
    C2W 23503967.037639
    L1C 123514198.424547
    L2W 96244784.438200
    S1C 44.187500
    S2W 43.375000
```

*Note: the keyword **last** substitutes an epoch selection with the last one in the container*

Remark: all the described outputs depend on actual verbosity settings

Console will alert when a key is not found:

```
> show obs MATE00ITA0 last
Too few arguments for keyword "obs". Should be "show_obs<site_epoch_sat>"
```

7.2 Data decoder monitoring

Similarly as browsing a data container described in Section 7.1, the console allows monitoring of individual decoding processes from data stream. For input decoder monitoring, the commands **list coder** and **set coder** allow listing and selecting a single decoder instance to work with:

```
> list coder
G-Nut console coders:
    (0) - path: ntrip://BCEP00BKGO/**:*:*:*:2101
    (1) - path: ntrip://MATE00ITA0/**:*:*:*:2101
    (2) - path: ntrip://WTZR00DEU0/**:*:*:*:2101
    (3) - path: ntrip://ZIM200CHE0/**:*:*:*:2101
> set coder 2
Selected coder: 2
```

Each decoder listed represents a single data stream that can be monitored. After selecting one, available commands can be printed with **show help**:

```
> show help
-----
Options
-----
    show          - monitoring decoding process
    show help     - this help
```

The monitoring of the selected decoder is triggered by the first **show** command. The console starts buffering all decoded messages of the input data stream. An execution of the command **show** prints decoded messages from the console internal buffer while cleaning reported data.

```

> verb 3
Selected verbosity: 3
> show

> show
Message: Reading RTCM message 1046
Message: Galileo broadcast ephemeris epoch decoded: 2021-11-29 12:50:00 (GAL) sat: E21
Message: Galileo broadcast ephemeris decoded:
E21 2021-11-29 12:50:00   INAV   759   759  93 107.00  000000000   1 1010000000   5   5
...

```

7.3 Repetitive command call

Using the **watch** mode, the console automatically calls selected command using a pre-defined interval (5 seconds by default). Currently, only the **show** command is allowed within a watch. A utilization is reported with the command **watch help**:

```

> watch help
-----
Watch help
-----
Periodically spawn a command. Just write "watch" before a command.
Currently supports 1 watch per client.
-----
Watch options
-----
watch + ...
      help      - this help
      stop      - end the watch
      int <secs> - set watch interval (default is 5s)
-----
Allowed commands
-----
watch + ...
      show ...

```

The **watch** mode doesn't block the console; therefore, a user may call another commands as needed. The **watch** also doesn't block any **exit** due to a user inactivity.

7.4 Application re-configuration

The console allows a modification of the software settings when using the **config** command followed by the command-line configuration arguments (these can also be chained) as follows:

```
config :out:verb=4 :gen:int=300
```

The **config** command without any argument prints the software current configuration.

The **reload** command enables reconfiguring the software by reading the initial XML configuration file (its path is fixed for the process and thus cannot be changed). Consequently, the original XML configuration file can be modified by user and re-read with the command in the console.

Bibliography

- [1] Zumberge J.F., Heffin M.B., Jefferson D.C., Watkins M.M., and Webb F.H. Precise point positioning for the efficient and robust analysis of GPS data from large networks. *Journal of Geophysics Research*, 102:5005–5018, 1997.
- [2] Vaclavovic P and Dousa J. Backward smoothing for precise GNSS applications. *Advances in space research*, 56(8):1627–1634, 2015.
- [3] IGS. International GNSS Service. <https://igs.org/formats-and-standards/>.
- [4] RTCM. Radio Technical Commission for Maritime Services. <https://rtcm.org/publications/>.
- [5] Mervart L. and Weber G. *BKG Ntrip Client*. Federal Agency for Cartography and Geodesy, Frankfurt, Germany.
- [6] Johnston G., Riddell A., and Hausler G. The International GNSS Service. In P. Teunissen and O. (Eds.) Montenbruck, editors, *Handbook of Global Navigation Satellite Systems (1st ed.)*, pages 967–982, Cham, Switzerland, 2017. Springer. doi:10.1007/978-3-319-42928-1.
- [7] Vaclavovic P., Dousa J., and Gyori G. G-Nut software library - state of development and first results. *Acta Geodynamica at Geomaterialia*, 10(4):431–436, 2013.
- [8] Hatanaka Y. A compression format and tools for GNSS observation data. *Bulletin of the Geospatial Information Authority of Japan*, 55(3):21–30, 2008. Available at <https://www.gsi.go.jp/ENGLISH/Bulletin55.html>.
- [9] Douša J. GOP’s consolidated multi-GNSS navigation data archive. available at <ftp://epncb.oma.be/pub/obs/BRDC/>, doi:10.24414/c4ba-kf16.
- [10] Saastamoinen J. Atmospheric correction for the troposphere and stratosphere in radio ranging of satellites. the use of artificial satellites for geodesy. *Geophys. Monogr.*, 15:247–251, 1972.
- [11] Böhm J., Heinkelmann R., and Schuh H. Short note: A global model of pressure and temperature for geodetic applications. *J Geod.*, 81:679–683, 2007.
- [12] H. S. Hopfield. Two-quartic tropospheric refractivity profile for correcting satellite data. *Journal of Geoph. Res*, 74:4487–4499, 1969.
- [13] Herring T. A. Shapiro I. I. Rogers A. E. E. Davis, J. L. and G. Elgered. Geodesy by radio interferometry: effects of atmospheric modelling errors on estimates of baseline length. *Radio Science*, 20:1593–1607, 1985.
- [14] RTCA. Minimum operational performance standards for Global Positioning System/Wide Area Augmentation System airborne equipment. *Radio Technical Commission for Aeronautics, SC-159*, 2006.
- [15] Tregoning P. Schuh H. Boehm J., Niell A. Global mapping function (gmf): A new empirical mapping function based on numerical weather model data. *Geophysical Research Letters*, 33, 2006.
- [16] Chen G. and Herring T.A. Effects of atmospheric azimuthal asymmetry on the analysis of space geodetic data. *Journal of Geof. Res*, 102(B9):20489–20502, 1997.

- [17] Borjesson J.A. Bar-Sever Y.E., Kroger P.M. Estimating horizontal gradients of tropospheric path delay with a single GPS receiver. *Journal of Geof. Res*, 103(B3):5019–5035, 1998.
- [18] McCarthy D. D. Iers conventions (1996). Technical report, Central Bureau of IERS - Observatoire de Paris, 1996.
- [19] Luzum B. Petit G. Iers conventions (2010). Technical report, Verlag des Bundesamts für Kartographie und Geodäsie, 2010.
- [20] Tung F. Striebel C.T. Rauch, H.E. Maximum likelihood estimates of linear dynamic systems. *AIAA Journal*, 3(8):1445–1450, 1965.
- [21] IERS. SINEX – Solution (Software/technique) INdependent EXchange Format Version 2.02. Technical report, 2006. available at <https://www.iers.org/IERS/EN/Organization/AnalysisCoordinator/SinexFormat/sinex.html>.
- [22] Pacione R. and Douša J. SINEX_TRO - Solution (Software/Technique) INdependent EXchange format for TROpospheric and meteorological parameters, 12 2020. https://files.igs.org/pub/data/format/sinex_tro_v2.00.pdf.
- [23] Vaclavovic P. and Dousa J. G-Nut/Anubis - open-source tool for multi-GNSS data monitoring with a multipath detection for new signals, frequencies and constellations. In *International Association of Geodesy Symposia*, volume 143. International Association of Geodesy, Springer, 2014.
- [24] Wu J., Wu S., Hajj G., Bertiguer W., and Lichten S. Effects of antenna orientation on gps carrier phase measurements. *Manuscripta Geodetica*, 18(2):91–98, 1993.
- [25] Böhm J., Möller G., Schindelegger M., Pain G., and Weber R. Development of an improved empirical model for slant delays in the troposphere (GPT2w). *GPS Solut.*, 19(3):433–441, 2014.
- [26] Martellucci A. Galileo reference troposphere model for the user receiver. *ESA-APPNG-REF/00621-AM v2.7*, 2012.