European Space Agency

Directorate of Operations and Infrastructure

NAPEOS

Mathematical Models and Algorithms

DOPS-SYS-TN-0100-OPS-GN

1.0

5-NOV-2009

ESOC European Space Operation Centre

DOCUMENT APPROVAL

Prepared by	*Organisation*	*Signature*	*Date*
T.A. Springer	PosiTim @ ESOC		5-Nov-2009

Approved by	*Organisation*	*Signature*	*Date*

Authorised by	*Organisation*	*Signature*	*Date*
J.M. Dow	ESOC		

CHANGE RECORD SHEET

Date	*Issue/Rev*	*Pages Affected*	*Description*	*DCR*	*Approval Authority*
5-Nov-2009	1.0	all	First ESOC issue		

TABLE OF CONTENTS

1 INTRODUCTION

1.1 Scope

NAPEOS (NAvigation Package for Earth Observation Satellites) is a portable navigation software system for Earth Observation satellite missions, providing orbit determination/prediction, manoeuvre-planning and parameter estimation capabilities and being able to process a wide variety of observation data including angles, range, range-rate, altimetry, satellite-to-satellite and Global Navigation Satellite System data. In particular, the following tracking systems are supported: GNSS (GPS, GLONASS, Galileo), satellite laser ranging (SLR), DORIS, PRARE, MPTS, and LCT ranging,

NAPEOS supports within ESOC all the activities in the International GNSS Service (IGS), International DORIS Service (IDS), and the International Laser Ranging Service (ILRS), and other international scientific communities. It is also used for the routine Envisat-1 POD evaluation.

NAPEOS will be operated through a graphical user interface. It will be portable to other UNIX environments (space agencies and other institutions).

1.2 Purpose

This document is a guide to the mathematical models and algorithms. Some models are taken form the literature without going further in the analysis of the theoretical assumptions. Other models have been developed from raw formulae, in order to fulfil the NAPEOS User Requirements. The description of algorithms and models reaches the software implementation level. However this is not a Detailed Design Document.

1.3 Structure of the document

This document is not structured ☺

1.4 Reference Documents

RD-1	F. Ceschino, J. Kuntzman. Numerical Solution of Initial Value Problems. Prentice-Hall, inc.1966
RD-2	Peter Henrici. Discrete Variable Methods in Ordinary Differential Equations.1962
RD-3	E. Fehlberg, NASA Technical Report TR R-287, 1968
RD-4	E.B.Shanks, Math.Comp. 20 (1966) PP. 21 - 38.
RD-5	B.D. Tapley et al. The Joint Gravity Model 3. Journal of Geophysical Research
RD-6	Frank D. Stacey. Physics of the Earth. Brookfield Press, 1992
RD-7	Dennis D. McCarthy and Gerard Petit (eds.) IERS Conventions (2003). IERS Technical note 32. Frankfurt am Main, Germany, 2004.
RD-8	Jean-Pierre Carrou (ed). Spaceflight Dynamics. CNES. 1995
RD-9	Loukis G. Agrotis. Determination of Satellite Orbits and the Global Positioning System, Thesis University of Nottingham, October 1984.
RD-10	H. F. Fliegel, T. E. Gallini, E. R. Swift. Global Positioning System Radiation Force Model For Geodetic Applications. J. Geophys. Res., 97
RD-11	S. J. Arnold, J. M. Dow. Models For Spacecraft Acceleration due to Earth Albedo and Infrared Radiation. OAD WP n. 265. June 1984
RD-12	M. Rosengren, ERS-1 - An Earth Observer that Exactly Follows Its Chosen Path, ESA Bulletin No. 72, Nov 1992
RD-13	M. Rosengren, The Orbit Control Of ERS-1, American Astronautical Society Publication, AAS-93-308, 1993
RD-14	M. Rosengren, Improved Technique for Passive Eccentricity Control, American Astronautical Society Publication, AAS-89-155, 1989
RD-15	L. Collatz, W. Wetterling, Optimization Problems, Springer-Verlag, 1975
RD-16	E. Kreyszig. Advance Engineering Mathematics. John Wiley & Sons, inc. 1993
RD-17	NAPEOS Technical Note 07. Target Definition and Orbit Control Feasibility.
RD-18	Optima Manual, Numerical Optimisation Centre, Hatfield Polytechnic, July 1989.

RD-19	PPF/Envisat 1 Orbit Corrections. Matra Marconi Space. PPF-MMB-TN-0588. January 1997
RD-20	Introducción a la Dinámica Espacial, T. Elices, INTA, 1991
RD-21	Spacecraft Attitude Determination and Control [Introduction to Estimation Theory, chapter 13.4, pp. 447-470], J. R. Wertz (ed.), Kluwer Academic Press, 1978.
RD-22	Factorization Methods for Discrete Sequential Estimation, G. J. Bierman, Academic Press, 1977.
RD-23	Applied Optimal Estimation, A. Gelb (ed.), The MIT Press, 1974.
RD-24	Theory of Satellite Geodesy, W. M. Kaula, Blaisdell Publishing Company, 1966.
RD-25	Fundamentals of Orbit Determination, B. D. Tapley, Lecture Notes in Earth Sciences No. 25, Springer-Verlag, 1989.
RD-26	Introduction to Estimation Theory, P. A. Pablos Chueca, GMVSA 2001/92, January 1992.
RD-27	Modifications to Programs BAHN and TRACK for Implementation of Orbit Determination Partial Derivatives by Numerical Integration of the Variational Equations, L. G. Agrotis, ESOC/OAD Software Description Document, December 1986.
RD-28	Precise Orbit Integration for Near-Earth Satellites, P. Duque, ESOC/OAD, October 1988.
RD-29	GEODYN System Description - Volume I, Goddard Space Flight Center, Greenbelt, Maryland, August 1976.Classical Fifth-, Sixth-, Seventh- and Eighth-Order Runge-Kutta Formulas with Stepsize Control, E. Fehlberg, NASA Technical Report 287, 1968.
RD-30	Spacecraft Attitude Determination and Control [Tracking ObservationsIntroduction to Estimation Theory, chapter 13.4, pp. 447-470],J. R. Wertz (ed.). Kluwer Academic Press, 1978.
RD-31	Applied Optimal Estimation, A. Gelb (ed.), The MIT Press, 1974.
RD-32	Ambigon, Blewitt
RD-33	Ge, M., Improving carrier-phase ambiguity resolution in global GPS network solutions, M. Ge et al., Journal of Geodesy, Volume 79, numbers 1-3, June 2005.
RD-34	Mervart, L., Ambiguity Resolution Techniques in Geodetic and Geodynamic Applications of the Global Positioning System, Band 53, Schweizerischen Geodätischen Kommission,Switzerland, 1995.
RD-35	Brockmann, E., Combination of Solutions for Geodetic and Geodynamic Applications of the Global Positioning System (GPS), Band 55, Schweizerischen Geodätischen Kommission, Switzerland, 1996.
RD-36	Springer, T.A., Modelling and Validating Orbits and Clocks Using the Global

	Positioning System, Bern, Switzerland, 1999.
RD-37	Saastamoinen, J., Atmospheric correction for the troposphere and stratosphere in radio ranging of satellites, in: Geophysical Monographs Series, 1972.
RD-38	Marini, J.W., and C.W. Murray, correction of Laser Range Tracking Data for Atmospheric Refraction at Elevations Above 10 degrees, X-591-73-351, NASA GSFC, 1973
RD-39	Mendes, V.B., and E. C. Pavlis (2004), High-Accuracy Zenith Delay Prediction at Optical Wavelengths, Geophys. Res. Lett., 31, L14602, doi:10.1029/2004GL020308.
RD-40	Niell AE (1996) Global Mapping Functions for the Atmosphere Delay at Radio Wavelength. Journal of Geoph. Research 101(B2), pp 3227-3246
RD-41	Boehm, J., R. Heinkelmann, and H. Schuh (2007). Short note: A global model of pressure and temperature for geodetic applications. Journal of Geodesy, Vol. 81, No. 10, pp. 679-683.
RD-42	Boehm J, AE Niell, P Tregoning, H Schuh (2006b) The Global Mapping Function (GMF): A new empirical mapping function based on data from numerical weather model data. Geophysical Research Letters, Vol. 33, L07304, doi:10.129/2005GL025546
RD-43	Leick, A., GPS Satellite Surveying, Wiley, ISBN 0-471-30626-6.

1.5 Acronyms

ADD	Architectural Design Document
DDD	Detailed Design Document
GLONASS	GLObal NAvigation Satellite System (Russia)
GNSS	Global Navigation Satellite System
GPS	Global Positioning System
URD	User Requirement Document

2 NAPEOS STANDARDS

The NAPEOS software relies heavily on the IERS Conventions 2003, RD-7 and follows these conventions very closely. However the IERS conventions do not provide a true standard as sometimes different options are given and allowed. So the purpose of this Chapter is to clearly identify and specify which part of the IERS Conventions 2003 was taken to clearly and unambiguously specify the standards used in NAPEOS. So to aid the reader the subsection of this Chapter follow the sections of the printed version of the IERS Conventions 2003.

2.1 General Definitions and Numerical Standards

NAPEOS follows the IERS Numerical standards as given in the Table 1.1.

However, the values in this table correspond to a so-called zero-tide system. NAPEOS, and in general satellite based techniques, works with a tide-free system. Consequently the following values were adapted to account for the different definitions:

Parameter	**IERS (zero-tide)**	**NAPEOS (tide-free)**
Ae: Equatorial Radius of the Earth	6378136.6 m	6378136.55 m
1/f: Flattening factor of the Earth	298.25642	298.25769
J2: Dynamical form factor	1.0826359×10^{-3}	1.0826267×10^{-3}
GM: Gravitational constant	$3.986004418 \times 10^{14}$ m^3/s^{-2}	$3.986004415 \times 10^{14}$ m^3/s^{-2}

The numerical standards in NAPEOS are handled by the module DBcb_Data.

The values of the velocity of light and GM may be found in the NAPEOS physical constants and central databases, respectively.

2.2 Conventional Celestial Reference System and Frame

Not used in NAPEOS.

2.3 Conventional Dynamical Realization of the ICRS

NAPEOS uses the DE405 planetary ephemeris to compute the positions of the Earth, Sun, Moon, and planets. However, the mass ratio w.r.t. the Earth and the radius are taken from the NAPEOS physical constants database. Those values are, however, in line with the values as specified for the DE405 planetary ephemerides.

2.4 Conventional Terrestrial Reference System and Frame

Besides by using the appropriate numerical standards the terrestrial reference frame is typically realised by using a set of well defined station coordinates. Currently, most of the tracking stations usually have coordinates given in the ITRF2005 reference frame.

2.5 Transformation Between the Celestial and Terrestrial Systems

NAPEOS strictly follows the IERS convention and used the IAU2000 transformation routines. The implemented method in NAPEOS is the method (1) which is the CEO-based transformations, using IAU 2000A precession-nutation. This uses the new (X, Y, s, Theta) transformation.

2.6 Geopotential

C_{21} and S_{21} computed based on mean pole offsets and the values of the adopted gravity field. Currently we typically use one of the latest Grace based gravity fields (e.g. EIGEN_GLO4)

Effect of solid earth tides modelled in full using the anelastic Earth model.

Solid Earth Pole tide accounted for as specified.

Ocean tides fully implemented. Currently using FES2004 based on GRACEERL03.

2.7 Displacements of Reference Points

Ocean loading values are obtained from the ocean loading service at:

http://www.oso.chalmers.se/~loading/

Currently we use the FES2004 with CMC corrections. Values have to be obtained for each site in the database. Missing values will cause a warning where needed.

The ocean loading constituents are used together with the IERS software (hardisp.f) to compute the actual station displacements.

Effect of solid Earth tides fully accounted for using the IERS software. In NAPEOS the software module is called tide2003.f.

Atmospheric loading is currently NOT implemented.

Deformation due to polar motion is implemented.

2.8 Tidal Variations in the Earth's Rotation

The tidal variations due to the Earth's rotations are fully implemented using the IERS subroutine ortho_eop.f. Furthermore, UT1UTC can not be interpolated linear over time spans of several hours because of the short periodic effects. Typically all short periodic effects, all periods up to 35 days, are removed from the UT1UTC values giving the so-called UT1UTC reduced curve. These values can be linearly interpolated. For this reduction a subroutine called Ut1r2003 was written which using the values in Table 8.1 of the IERS Conventions (2003) for periods up to 35 days.

2.9 Tropospheric Model

For SLR the latest model from Pavlis and Mendes, RD-39, is implemented. But also the "old" Marinni Murray model, RD-38, is still available.

For GNSS processing different models are available.

Available options for obtaining apriori models for temperature and pressure

- Take from the input observations (e.g. for SLR)
- Standard atmosphere (one temperature, pressure and humidity for all points on and near the earth)
- Height dependent standard atmosphere (to be implemented)
- GPT model (based on gpt.f software from J. Boehm RD-41)

Available options for computing a priori zenith delay:

- Saastamoinen, RD-37

Available options for mapping a priori zenith delay

- Saastamoinen, RD-37
- Niell mapping function (NMF, RD-40), wet or dry
- Global mapping function (GMF, RD-42), wet/hydrostatic or dry

Available options for mapping the estimated zenith delay

- Niell mapping function (NMF, RD-40), wet or dry
- Global mapping function (GMF, RD-42), wet/hydrostatic or dry

2.10 General Relativistic Models for equations of motion

Relativistic acceleration of a artificial Earth Satellite

Lense-Thirring and de Sitter corrections ignored

2.11 General Relativistic Models for Propagation

The correction denoted for “laser ranging” is applied to all observations.

For GPS also the relativistic effect on the satellite clocks due to the eccentricity of the satellite orbit is taken into account.

3 OBSERVATION EQUATIONS

3.1 Definitions

- Satellite position and velocity : $\vec{x}_{sat}$, $\vec{v}_{sat}$
- Station position and velocity: $\vec{x}_{sta}$, $\vec{v}_{sta}$
- Generically, emitter position and velocity : $\vec{x}_e$, $\vec{v}_e$
- Generically, receiver position and velocity: $\vec{x}_r$, $\vec{v}_r$
- Station to satellite relative position and velocity: $\Delta\vec{x}$, $\Delta\vec{v}$

  Satellite and station positions are always referred to as inertial. Only in the partial derivatives of observation with respect to station positions are referred to the Earth fixed reference frame.
- Geometric range: $\rho = |\Delta\vec{x}|$
- Geometric range rate: $\dot{\rho} = \frac{d\rho}{dt}$
- Rotation matrix from inertial to topocentric and its time derivative: $T, \dot{T}$
- Rotation matrix from inertial to Earth fixed and its time derivative: $E, \dot{E}$
- Topocentric reference frame unit vectors and their time derivatives (east, north, up): $(\vec{e}, \vec{n}, \vec{u})$, $(\dot{\vec{e}}, \dot{\vec{n}}, \dot{\vec{u}})$
- Geometric observation: y_g
- Geometric observation corrected for time of flight delay, time bias and corrections : y_c
- Reference time correlation (multiplied by the Reference time correlation function Ψ defined below to give the time of flight observation correction): Δt_{ref}
- Time tag bias: Δt_{tag}, convention is $t_{clock} = t_{UTC} + \Delta t_{tag}$. This is the time bias of the element in the observation that sets the tag to the observation. If there are two elements in the observation, the other will be considered as Δt_{no_tag}
- Measurement corrections (bias, transponder delay, atmospheric delay, etc., consistent with the type of observation, i.e. two way observation implies this is two way correction): $\Delta\rho$, $\Delta\dot{\rho}$

- Emitter and receiver time biases: Δt_e, Δt_r (when the observation is referred to stations and satellites rather than emitter and receiver, these become Δt_{stat}, Δt_{sat} as appropriate for the type of observation)
- Speed of light: c
- Derivative w.r.t. a vector: $\frac{\partial y}{\partial \vec{x}} = \begin{bmatrix} \frac{\partial y}{\partial x_1} \\ \frac{\partial y}{\partial x_2} \\ \frac{\partial y}{\partial x_3} \end{bmatrix}$
- Reference time tag correlation function: $\Psi = \begin{cases} 0, \text{ No event time reference correction} \\ -1, \text{ Event time is tracker receive time} \\ 0, \text{ Event time is tracked trasmit time} \\ +1, \text{ Event time is tracker transmit time} \\ 0, \text{ Event time is tracked receive time} \end{cases}$
- Vector rotation convention:

$$\vec{x}_{In} = M \cdot \vec{x}_{EF} \quad \Rightarrow \quad (\vec{x}_I)_i = \sum_{j=1}^{n} m_{ij} \cdot (X_{EF})_j \quad , \qquad m_{ij} \equiv m(row\ i, column\ j)$$

3.2 Ground station to satellite 1-Way Range (pseudo range)

Geometric observation

$$\Delta\vec{x} = \vec{x}_e - \vec{x}_r , \qquad \Delta\vec{v} = \vec{v}_e - \vec{v}_r$$

$$\rho = \left|\Delta\vec{x}\right| , \qquad \dot{\rho} = \frac{\Delta\vec{v}\cdot\Delta\vec{x}}{\rho}$$

$$y_g = \rho$$

Observation partial derivatives w.r.t. satellite and station position and velocity

$$\frac{\partial y_g}{\partial\vec{x}_{sat}} = \frac{\Delta\vec{x}}{\rho}\text{, for emitter satellite} \qquad \frac{\partial y_g}{\partial\vec{x}_{sat}} = -\frac{\Delta\vec{x}}{\rho}\text{, for receiver satellite} \qquad \frac{\partial y_g}{\partial\vec{v}_{sat}} = 0$$

$$\frac{\partial y_g}{\partial\vec{x}_{sta}} = -E\frac{\Delta\vec{x}}{\rho}\text{, for receiver station} \qquad \frac{\partial y_g}{\partial\vec{x}_{sta}} = E\frac{\Delta\vec{x}}{\rho}\text{, for emitter station} \qquad \frac{\partial y_g}{\partial\vec{v}_{sta}} = 0$$

Observation partial derivative w.r.t. to time

$$\frac{\partial y_g}{\partial t} = \dot{\rho}$$

Computed observation

$$y_c = y_g - \frac{\partial y_g}{\partial t}\cdot\frac{\Delta\tilde{\rho}_{tag}}{c} - \frac{\vec{v}_{no\_tag}\cdot\Delta\vec{x}}{c} + \Delta\tilde{\rho}_e + \Delta\tilde{\rho}_r\text{, if observation is tagged at reception time}$$

$$y_c = y_g + \frac{\partial y_g}{\partial t}\cdot\frac{\Delta\tilde{\rho}_{tag}}{c} - \frac{\vec{v}_{no\_tag}\cdot\Delta\vec{x}}{c} + \Delta\tilde{\rho}_e + \Delta\tilde{\rho}_r\text{, if observation is tagged at emission time}$$

where:

$$\Delta\tilde{\rho}_{tag} = c\Delta t_r + \Delta\rho_r\text{, if observation is tagged at reception time}$$

$$\Delta\tilde{\rho}_{tag} = c\Delta t_e - \Delta\rho_e\text{, if observation is tagged at emission time}$$

3.3 Ground station to satellite 2-Way Range

Geometric observation

$$\Delta\vec{x} = \vec{x}_e - \vec{x}_r\,, \qquad \Delta\vec{v} = \vec{v}_e - \vec{v}_r$$

$$\rho = \left|\Delta\vec{x}\right|, \qquad \dot{\rho} = \frac{\Delta\vec{v}\cdot\Delta\vec{x}}{\rho}$$

$$y_g = 2\rho$$

Observation partial derivatives w.r.t. satellite and station position and velocity

$$\frac{\partial y_g}{\partial \vec{x}_{sat}} = 2\frac{\Delta\vec{x}}{\rho}\text{, for emitter satellite} \qquad \frac{\partial y_g}{\partial \vec{x}_{sat}} = -2\frac{\Delta\vec{x}}{\rho}\text{, for receiver satellite} \qquad \frac{\partial y_g}{\partial \vec{v}_{sat}} = 0$$

$$\frac{\partial y_g}{\partial \vec{x}_{sta}} = -2E\frac{\Delta\vec{x}}{\rho}\text{, for receiver station} \qquad \frac{\partial y_g}{\partial \vec{x}_{sta}} = 2E\frac{\Delta\vec{x}}{\rho}\text{, for emitter station} \qquad \frac{\partial y_g}{\partial \vec{v}_{sta}} = 0$$

Observation partial derivative w.r.t. to time

$$\frac{\partial y_g}{\partial t} = 2\dot{\rho}$$

Observation time tag correlation

$$\Delta t_{ref} = \Psi\frac{\rho + \frac{1}{2}\left(\Delta\rho_e + \Delta\rho_r\right)}{c}$$

Computed observation

$$y_c = y_g + \left(\Delta t_{ref} - \Delta t_{tag}\right)\frac{\partial y_g}{\partial t} + \Delta\rho_r + \Delta\rho_e$$

3.4 Station to satellite to second station and back 4-Way Range

Geometric observation

$$\Delta\vec{x}_1 = \vec{x}_{sat} - \vec{x}_{sta1}, \qquad \Delta\vec{v}_1 = \vec{v}_{sat} - \vec{v}_{sta1}$$

$$\Delta\vec{x}_2 = \vec{x}_{sat} - \vec{x}_{sta2}, \qquad \Delta\vec{v}_2 = \vec{v}_{sat} - \vec{v}_{sta2}$$

$$\rho_1 = \left|\Delta\vec{x}_1\right|, \qquad \dot{\rho}_1 = \frac{\Delta\vec{v}_1 \cdot \Delta\vec{x}_1}{\rho_1}$$

$$\rho_2 = \left|\Delta\vec{x}_2\right|, \qquad \dot{\rho}_2 = \frac{\Delta\vec{v}_2 \cdot \Delta\vec{x}_2}{\rho_2}$$

$$y_g = 2\rho_1 + 2\rho_2$$

Observation partial derivatives w.r.t. satellite position and velocity

$$\frac{\partial y_g}{\partial \vec{x}_{sat}} = 2\frac{\Delta\vec{x}_1}{\rho_1} + 2\frac{\Delta\vec{x}_2}{\rho_2}, \quad \frac{\partial y_g}{\partial \vec{v}_{sat}} = 0$$

Observation partial derivatives w.r.t. first and second stations position and velocity

$$\frac{\partial y_g}{\partial \vec{x}_{sta1}} = -2E_1\frac{\Delta\vec{x}_1}{\rho_1}, \qquad \frac{\partial y_g}{\partial \vec{v}_{sta1}} = 0$$

$$\frac{\partial y_g}{\partial \vec{x}_{sta2}} = -2E_2\frac{\Delta\vec{x}_2}{\rho_2}, \qquad \frac{\partial y_g}{\partial \vec{v}_{sta2}} = 0$$

Observation partial derivative w.r.t. to time

$$\frac{\partial y_g}{\partial t} = 2\dot{\rho}_1 + 2\dot{\rho}_2$$

Observation time tag correlation

$$\Delta t_{ref} = \Psi\frac{\rho_1 + \rho_2 + \frac{1}{2}\left(\Delta\rho_{stat1} + \Delta\rho_{stat2} + \Delta\rho_{sat}\right)}{c}$$

Computed observation

$$y_c = y_g + \left(\Delta t_{tag} + \Delta t_{ref}\right)\frac{\partial y_g}{\partial t} + \Delta\rho_{stat1} + \Delta\rho_{stat2} + \Delta\rho_{sat}$$

3.5 Ground station to satellite 1-Way Range Rate

Geometric observation

$$\Delta\vec{x} = \vec{x}_{sat} - \vec{x}_{sta}\,, \qquad \Delta\vec{v} = \vec{v}_{sat} - \vec{v}_{sta}$$

$$\rho = \left|\Delta\vec{x}\right|, \qquad \dot{\rho} = \frac{\Delta\vec{v}\cdot\Delta\vec{x}}{\rho}$$

$$y_g = \dot{\rho}$$

Observation partial derivatives w.r.t. satellite and station position and velocity

$$\frac{\partial y_g}{\partial\vec{x}_{sat}} = \frac{\Delta\vec{v} - \dot{\rho}\dfrac{\Delta\vec{x}}{\rho}}{\rho}, \qquad \frac{\partial y_g}{\partial\vec{v}_{sat}} = \frac{\Delta\vec{x}}{\rho}$$

$$\frac{\partial y_g}{\partial\vec{x}_{sta}} = \frac{1}{\rho}\left[E\left(\dot{\rho}\frac{\Delta\vec{x}}{\rho} - \Delta\vec{v}\right) - \dot{E}\Delta\vec{x}\right], \qquad \frac{\partial y_g}{\partial\vec{v}_{sta}} = 0$$

Observation partial derivative w.r.t. to time

$$\frac{\partial y_g}{\partial t} = \frac{1}{\rho}\left[\left(\Delta\vec{v}\right)^2 - \Delta\vec{x}\cdot\left(\frac{\mu}{r^3}\vec{x}_{sat} - \overline{\Omega}_{\oplus}\times\vec{v}_{stat}\right) - \dot{\rho}^2\right]$$

Observation time tag correlation

$$\Delta t_{ref} = \Psi\frac{\rho}{c}$$

Computed observation

$$y_c = y_g + \left(\Delta t_{tag} + \Delta t_{ref}\right)\frac{\partial y_g}{\partial t} + \Delta\dot{\rho}_{stat} + \Delta\dot{\rho}_{sat}$$

3.6 Station to satellite to second satellite and back 4-Way Range

Geometric observation

$$\Delta\vec{x}_1 = \vec{x}_{sat1} - \vec{x}_{sta}, \qquad \Delta\vec{v}_1 = \vec{v}_{sat1} - \vec{v}_{sta}$$

$$\Delta\vec{x}_2 = \vec{x}_{sat1} - \vec{x}_{sat2}, \qquad \Delta\vec{v}_2 = \vec{v}_{sat1} - \vec{v}_{sat2}$$

$$\rho_1 = \left|\Delta\vec{x}_1\right|, \qquad \dot{\rho}_1 = \frac{\Delta\vec{v}_1 \cdot \Delta\vec{x}_1}{\rho_1}$$

$$\rho_2 = \left|\Delta\vec{x}_2\right|, \qquad \dot{\rho}_2 = \frac{\Delta\vec{v}_2 \cdot \Delta\vec{x}_2}{\rho_2}$$

$$y_g = 2\rho_1 + 2\rho_2$$

Observation partial derivatives w.r.t. first and second satellites position and velocity

$$\frac{\partial y_g}{\partial \vec{x}_{sat1}} = 2\frac{\Delta\vec{x}_1}{\rho_1} + 2\frac{\Delta\vec{x}_2}{\rho_2}, \qquad \frac{\partial y_g}{\partial \vec{v}_{sat}} = 0$$

$$\frac{\partial y_g}{\partial \vec{x}_{sat2}} = -2\frac{\Delta\vec{x}_2}{\rho_2}, \qquad \frac{\partial y_g}{\partial \vec{v}_{sta2}} = 0$$

Observation partial derivatives w.r.t. station position and velocity

$$\frac{\partial y_g}{\partial \vec{x}_{sta}} = -2E_1\frac{\Delta\vec{x}_1}{\rho_1}, \qquad \frac{\partial y_g}{\partial \vec{v}_{sta}} = 0$$

Observation partial derivative w.r.t. to time

$$\frac{\partial y_g}{\partial t} = 2\dot{\rho}_1 + 2\dot{\rho}_2$$

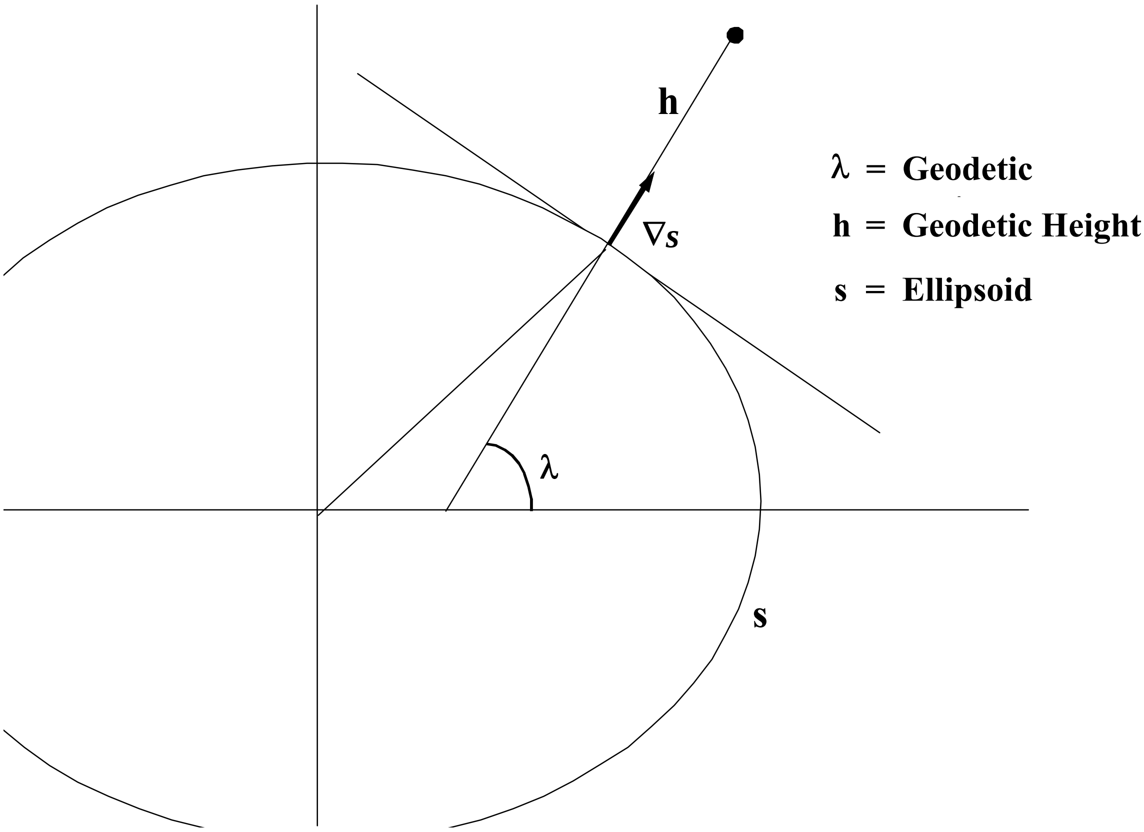
Observation time tag correlation

$$\Delta t_{ref} = \Psi\frac{\rho_1 + \rho_2 + \frac{1}{2}\left(\Delta\rho_{stat} + \Delta\rho_{sat1} + \Delta\rho_{sat2}\right)}{c}$$

Computed observation

$$y_c = y_g + \left(\Delta t_{tag} + \Delta t_{ref}\right)\frac{\partial y_g}{\partial t} + \Delta\rho_{stat} + \Delta\rho_{sat1} + \Delta\rho_{sat2}$$

3.7 Satellite Altimetry



Geometric observation

$y_g = h = \vec{x}_{sat} - \vec{x}_{subsat}$ (this is the geodetic height)

Observation partial derivatives w.r.t. satellite position and velocity

$$\nabla s = \frac{\vec{x}_{sat} - \vec{x}_{subsat}}{h}$$ (gradient of the ellipsoid surface in inertial coordinates)

Observation partial derivative w.r.t. to time

$$\nabla s = \frac{\vec{x}_{sat} - \vec{x}_{subsat}}{h}$$ (gradient of the ellipsoid surface in inertial coordinates)

$$\frac{\partial y_g}{\partial \vec{x}_{sat}} = \nabla s\,, \qquad \frac{\partial y_g}{\partial \vec{v}_{sat}} = 0$$

Observation partial derivative w.r.t. to time

$$\frac{\partial y_g}{\partial t} = \dot{h} = \vec{v}_{sat} \cdot \nabla s$$

Observation time tag correlation

$$\Delta t_{ref} = \Psi \frac{\rho + \Delta\rho}{c}$$

Computed observation

$$y_c = y_g + \left(\Delta t_{tag} + \Delta t_{ref}\right)\frac{\partial y_g}{\partial t} + \Delta\rho$$

3.8 Satellite Inertial position

Geometric observation

$$y_g^i = x_{sat}^i, \qquad \mathrm{i} = 1,2,3 \qquad \text{(in inertial reference frame)}$$

Observation partial derivatives w.r.t. satellite position

$$\frac{\partial y_g^i}{\partial x_{sat}^j} = \delta_i^j, \qquad \mathrm{i, j} = 1,2,3 \qquad \text{where} \qquad \delta_i^j = \begin{cases} 1, \mathrm{i} = \mathrm{j} \\ 0, \mathrm{i} \neq \mathrm{j} \end{cases} \text{(Kronecker symbol)}$$

Observation partial derivative w.r.t. time

$$\frac{\partial y_g^i}{\partial t} = v_{sat}^i, \qquad \mathrm{i} = 1,2,3 \qquad \text{(in inertial reference frame)}$$

Computed observation

$$y_c = y_g + \Delta t_{tag} \frac{\partial y_g}{\partial t}$$

3.9 Satellite Earth Fixed position

Geometric observation

$y_g^i = x_{sat}^i$, i = 1,2,3 (in Earth-fixed reference frame)

Observation partial derivatives w.r.t. satellite position

$E = e_{ij}$, $\vec{x}_I = E \cdot \vec{x}_{EF}$, (rotation matrix from Earth-fixed to inertial frame)

$$\frac{\partial y_g^i}{\partial x_{sat}^j} = e_{ij} \quad \Rightarrow \quad \frac{\partial y_g^i}{\partial \vec{x}_{sat}} = \begin{bmatrix} e_{1i} \\ e_{2i} \\ e_{3i} \end{bmatrix}, \qquad \text{i, j} = 1,2,3$$

Observation partial derivative w.r.t. time

$$\frac{\partial y_g^i}{\partial t} = v_{sat}^i, \qquad \text{i} = 1,2,3 \qquad \text{(in Earth-fixed reference frame)}$$

Computed observation

$$y_c = y_g + \Delta t_{tag} \frac{\partial y_g}{\partial t}$$

3.10 Satellite to Satellite 1-Way range (satellite to satellite pseudo-range)

Geometric observation

$$\Delta\vec{x} = \vec{x}_e - \vec{x}_r, \qquad \Delta\vec{v} = \vec{v}_e - \vec{v}_r$$

$$\rho = \left|\Delta\vec{x}\right|, \qquad \dot{\rho} = \frac{\Delta\vec{v}\cdot\Delta\vec{x}}{\rho}$$

$$y_g = \rho$$

Observation partial derivatives w.r.t. emitter satellite position and velocity

$$\frac{\partial y_g}{\partial \vec{x}_e} = \frac{\Delta\vec{x}}{\rho}, \qquad \frac{\partial y_g}{\partial \vec{v}_e} = 0$$

Observation partial derivatives w.r.t. receiver satellite position and velocity

$$\frac{\partial y_g}{\partial \vec{x}_r} = -\frac{\Delta\vec{x}}{\rho}, \qquad \frac{\partial y_g}{\partial \vec{v}_r} = 0$$

Observation partial derivative w.r.t. to time

$$\frac{\partial y_g}{\partial t} = \dot{\rho}$$

Computed observation

$$y_c = y_g - \frac{\partial y_g}{\partial t}\cdot\frac{\Delta\tilde{\rho}_{tag}}{c} - \frac{\vec{v}_{no\_tag}\cdot\Delta\vec{x}}{c} + \Delta\tilde{\rho}_e + \Delta\tilde{\rho}_r \text{, if observation is tagged at reception time}$$

$$y_c = y_g + \frac{\partial y_g}{\partial t}\cdot\frac{\Delta\tilde{\rho}_{tag}}{c} - \frac{\vec{v}_{no\_tag}\cdot\Delta\vec{x}}{c} + \Delta\tilde{\rho}_e + \Delta\tilde{\rho}_r \text{, if observation is tagged at emission time}$$

where:

$$\Delta\tilde{\rho}_{tag} = c\Delta t_r + \Delta\rho_r \text{, if observation is tagged at reception time}$$

$$\Delta\tilde{\rho}_{tag} = c\Delta t_e - \Delta\rho_e \text{, if observation is tagged at emission time}$$

3.11 Satellite to satellite 2-Way Range

Geometric observation

$$\Delta\vec{x} = \vec{x}_e - \vec{x}_r\,, \qquad \Delta\vec{v} = \vec{v}_e - \vec{v}_r$$

$$\rho = \left|\Delta\vec{x}\right|, \qquad \dot{\rho} = \frac{\Delta\vec{v}\cdot\Delta\vec{x}}{\rho}$$

$$y_g = 2\rho$$

Observation partial derivatives w.r.t. emitter satellite position and velocity

$$\frac{\partial y_g}{\partial \vec{x}_e} = 2\frac{\Delta\vec{x}}{\rho}\,, \qquad \frac{\partial y_g}{\partial \vec{v}_e} = 0$$

Observation partial derivatives w.r.t. receiver satellite position and velocity

$$\frac{\partial y_g}{\partial \vec{x}_r} = -2\frac{\Delta\vec{x}}{\rho}\,, \qquad \frac{\partial y_g}{\partial \vec{v}_r} = 0$$

Observation partial derivative w.r.t. to time

$$\frac{\partial y_g}{\partial t} = 2\dot{\rho}$$

Observation time tag correlation

$$\Delta t_{ref} = \Psi\frac{\rho + \frac{1}{2}\left(\Delta\rho_e + \Delta\rho_r\right)}{c}$$

Computed observation

$$y_c = y_g + \left(\Delta t_{ref} - \Delta t_{tag}\right)\frac{\partial y_g}{\partial t} + \Delta\rho_r + \Delta\rho_e$$

3.12 Satellite to Satellite 1-Way Range Rate

Geometric observation

$$\Delta\vec{x} = \vec{x}_r - \vec{x}_e, \qquad \Delta\vec{v} = \vec{v}_r - \vec{v}_e$$

$$\rho = \left|\Delta\vec{x}\right|, \qquad \dot{\rho} = \frac{\Delta\vec{v}\cdot\Delta\vec{x}}{\rho}$$

$$y_g = \dot{\rho}$$

Observation partial derivatives w.r.t. emitter satellite position and velocity

$$\frac{\partial y_g}{\partial \vec{x}_e} = -\frac{\Delta\vec{v} - \dot{\rho}\dfrac{\Delta\vec{x}}{\rho}}{\rho}, \qquad \frac{\partial y_g}{\partial \vec{v}_e} = -\frac{\Delta\vec{x}}{\rho}$$

Observation partial derivatives w.r.t. receiver satellite position and velocity

$$\frac{\partial y_g}{\partial \vec{x}_r} = \frac{\Delta\vec{v} - \dot{\rho}\dfrac{\Delta\vec{x}}{\rho}}{\rho}, \qquad \frac{\partial y_g}{\partial \vec{v}_r} = \frac{\Delta\vec{x}}{\rho}$$

Observation partial derivative w.r.t. to time

$$\frac{\partial y_g}{\partial t} = \frac{1}{\rho}\left[\left(\Delta\vec{v}\right)^2 - \Delta\vec{x}\cdot\mu\left(\frac{\vec{x}_e}{r_e^{\,3}} - \frac{\vec{x}_r}{r_r^{\,3}}\right) - \dot{\rho}^2\right]$$

Observation time tag correlation

$$\Delta t_{ref} = \Psi\frac{\rho}{c}$$

Computed observation

$$y_c = y_g + \left(\Delta t_{tag} + \Delta t_{ref}\right)\frac{\partial y_g}{\partial t} + \Delta\dot{\rho}_{sat1} + \Delta\dot{\rho}_{sat2}$$

3.13 Satellite to Satellite 2-Way Range Rate

Geometric observation

$$\Delta\vec{x} = \vec{x}_r - \vec{x}_e, \quad \Delta\vec{v} = \vec{v}_r - \vec{v}_e$$

$$\rho = |\Delta\vec{x}|, \qquad \dot{\rho} = \frac{\Delta\vec{v}\cdot\Delta\vec{x}}{\rho}$$

$$y_g = 2\dot{\rho}$$

Observation partial derivatives w.r.t. emitter satellite position and velocity

$$\frac{\partial y_g}{\partial \vec{x}_e} = -2\frac{\Delta\vec{v} - \dot{\rho}\dfrac{\Delta\vec{x}}{\rho}}{\rho}, \quad \frac{\partial y_g}{\partial \vec{v}_e} = -2\frac{\Delta\vec{x}}{\rho}$$

Observation partial derivatives w.r.t. receiver satellite position and velocity

$$\frac{\partial y_g}{\partial \vec{x}_r} = 2\frac{\Delta\vec{v} - \dot{\rho}\dfrac{\Delta\vec{x}}{\rho}}{\rho}, \quad \frac{\partial y_g}{\partial \vec{v}_r} = 2\frac{\Delta\vec{x}}{\rho}$$

Observation partial derivative w.r.t. to time

$$\frac{\partial y_g}{\partial t} = \frac{2}{\rho}\left[(\Delta\vec{v})^2 - \Delta\vec{x}\cdot\mu\left(\frac{\vec{x}_e}{r_e^{\,3}} - \frac{\vec{x}_r}{r_r^{\,3}}\right) - \dot{\rho}^2\right]$$

Observation time tag correlation

$$\Delta t_{ref} = \Psi\frac{\rho}{c}$$

Computed observation

$$y_c = y_g + (\Delta t_{tag} + \Delta t_{ref})\frac{\partial y_g}{\partial t} + \Delta\dot{\rho}_{sat1} + \Delta\dot{\rho}_{sat2}$$

3.14 GNSS Double Differences between two Ground Stations

GNSS Satellites : 1, 2

GNSS Ground Stations : a, b. According to Napeos conventions, Station b is always the MASTER station in the double difference generation.

Geometric observation

$$\Delta\vec{x}_{i\alpha} = \vec{x}_i - \vec{x}_\alpha , \qquad \Delta\vec{v}_{i\alpha} = \vec{v}_i - \vec{v}_\alpha , \qquad \left(\begin{array}{l} i = 1,2 \\ \alpha = \mathrm{a,b} \end{array} \right.$$

$$\rho_{i\alpha} = \left| \Delta\vec{x}_{i\alpha} \right| , \qquad \dot{\rho}_{i\alpha} = \frac{\Delta\vec{v}_{i\alpha} \cdot \Delta\vec{x}_{i\alpha}}{\rho_{i\alpha}} , \qquad \left(\begin{array}{l} i = 1,2 \\ \alpha = \mathrm{a,b} \end{array} \right.$$

$$y_g = \rho_{1a} - \rho_{1b} - (\rho_{2a} - \rho_{2b})$$

Observation partial derivatives w.r.t. first and second GNSS satellite position and velocity

$$\frac{\partial y_g}{\partial \vec{x}_1} = \frac{\Delta\vec{x}_{1a}}{\rho_{1a}} - \frac{\Delta\vec{x}_{1b}}{\rho_{1b}} , \qquad \frac{\partial y_g}{\partial \vec{v}_1} = 0$$

$$\frac{\partial y_g}{\partial \vec{x}_2} = \frac{\Delta\vec{x}_{2b}}{\rho_{2b}} - \frac{\Delta\vec{x}_{2a}}{\rho_{2a}} , \qquad \frac{\partial y_g}{\partial \vec{v}_2} = 0$$

Observation partial derivatives w.r.t. first and second stations position and velocity

$$\frac{\partial y_g}{\partial \vec{x}_a} = E_a \left(\frac{\Delta\vec{x}_{2a}}{\rho_{2a}} - \frac{\Delta\vec{x}_{1a}}{\rho_{1a}} \right) , \qquad \frac{\partial y_g}{\partial \vec{v}_a} = 0$$

$$\frac{\partial y_g}{\partial \vec{x}_b} = E_b \left(\frac{\Delta\vec{x}_{1b}}{\rho_{1b}} - \frac{\Delta\vec{x}_{2b}}{\rho_{2b}} \right) , \qquad \frac{\partial y_g}{\partial \vec{v}_b} = 0$$

Observation partial derivative w.r.t. to time

$$\frac{\partial y_g}{\partial t} = \dot{\rho}_{1a} - \dot{\rho}_{1b} - (\dot{\rho}_{2a} - \dot{\rho}_{2b})$$

Observation time tag correlation

All ranges (ρ) are computed using the correction in the *Station-to-Satellite* ***1-way range*** algorithm described above.

Computed observation

$$y_c = y_g$$

3.15 GNSS Double Differences between a Ground Station and an Orbiting Receiver

GNSS Satellites : 1, 2

GNSS Ground Station : a

GNSS Orbiting Receiver : b.

According to Napeos conventions, the Orbiting Receiver (b) is always the MASTER in the double difference generation

Geometric observation

$$\Delta\vec{x}_{i\alpha} = \vec{x}_i - \vec{x}_\alpha\,, \qquad \Delta\vec{v}_{i\alpha} = \vec{v}_i - \vec{v}_\alpha\,, \qquad \left(\begin{array}{l} i = 1{,}2 \\ \alpha = \mathrm{a,b} \end{array}\right.$$

$$\rho_{i\alpha} = \left|\Delta\vec{x}_{i\alpha}\right|, \qquad \dot{\rho}_{i\alpha} = \frac{\Delta\vec{v}_{i\alpha}\cdot\Delta\vec{x}_{i\alpha}}{\rho_{i\alpha}}, \qquad \left(\begin{array}{l} i = 1{,}2 \\ \alpha = \mathrm{a,b} \end{array}\right.$$

$$y_g = \rho_{1a} - \rho_{1b} - \left(\rho_{2a} - \rho_{2b}\right)$$

Observation partial derivatives w.r.t. first and second GNSS satellite position and velocity

$$\frac{\partial y_g}{\partial\vec{x}_1} = \frac{\Delta\vec{x}_{1a}}{\rho_{1a}} - \frac{\Delta\vec{x}_{1b}}{\rho_{1b}}, \qquad \frac{\partial y_g}{\partial\vec{v}_1} = 0$$

$$\frac{\partial y_g}{\partial\vec{x}_2} = \frac{\Delta\vec{x}_{2b}}{\rho_{2b}} - \frac{\Delta\vec{x}_{2a}}{\rho_{2a}}, \qquad \frac{\partial y_g}{\partial\vec{v}_2} = 0$$

Observation partial derivatives w.r.t. ground station position and velocity

$$\frac{\partial y_g}{\partial\vec{x}_a} = E_a\left(\frac{\Delta\vec{x}_{2a}}{\rho_{2a}} - \frac{\Delta\vec{x}_{1a}}{\rho_{1a}}\right), \qquad \frac{\partial y_g}{\partial\vec{v}_a} = 0$$

Observation partial derivatives w.r.t. orbiting receiver position and velocity

$$\frac{\partial y_g}{\partial\vec{x}_b} = \frac{\Delta\vec{x}_{1b}}{\rho_{1b}} - \frac{\Delta\vec{x}_{2b}}{\rho_{2b}}, \qquad \frac{\partial y_g}{\partial\vec{v}_b} = 0$$

Observation partial derivative w.r.t. to time

$$\frac{\partial y_g}{\partial t} = \dot{\rho}_{1a} - \dot{\rho}_{1b} - \left(\dot{\rho}_{2a} - \dot{\rho}_{2b}\right)$$

Observation time tag correlation

All ranges (ρ) are computed using the correction in the *Station-to-Satellite and Satellite-to-Satellite* ***1-way range*** algorithms described above.

Computed observation

$$y_c = y_g$$

3.16 GNSS Double Differences between two Orbiting Receivers

GNSS Satellites : 1, 2

GNSS Orbiting Receivers : a, b.

Geometric observation

$$\Delta\vec{x}_{i\alpha} = \vec{x}_i - \vec{x}_\alpha\,, \qquad \Delta\vec{v}_{i\alpha} = \vec{v}_i - \vec{v}_\alpha\,, \qquad \left(\begin{array}{l} i = 1{,}2 \\ \alpha = \mathrm{a,b} \end{array}\right.$$

$$\rho_{i\alpha} = \left|\Delta\vec{x}_{i\alpha}\right|, \qquad \dot{\rho}_{i\alpha} = \frac{\Delta\vec{v}_{i\alpha}\cdot\Delta\vec{x}_{i\alpha}}{\rho_{i\alpha}}, \qquad \left(\begin{array}{l} i = 1{,}2 \\ \alpha = \mathrm{a,b} \end{array}\right.$$

$$y_g = \rho_{1a} - \rho_{1b} - \left(\rho_{2a} - \rho_{2b}\right)$$

Observation partial derivatives w.r.t. first and second GNSS satellite position and velocity

$$\frac{\partial y_g}{\partial\vec{x}_1} = \frac{\Delta\vec{x}_{1a}}{\rho_{1a}} - \frac{\Delta\vec{x}_{1b}}{\rho_{1b}}, \qquad \frac{\partial y_g}{\partial\vec{v}_1} = 0$$

$$\frac{\partial y_g}{\partial\vec{x}_2} = \frac{\Delta\vec{x}_{2b}}{\rho_{2b}} - \frac{\Delta\vec{x}_{2a}}{\rho_{2a}}, \qquad \frac{\partial y_g}{\partial\vec{v}_2} = 0$$

Observation partial derivatives w.r.t. first and second orbiting receivers position and velocity

$$\frac{\partial y_g}{\partial\vec{x}_a} = \frac{\Delta\vec{x}_{1a}}{\rho_{1a}} - \frac{\Delta\vec{x}_{1b}}{\rho_{1b}}, \qquad \frac{\partial y_g}{\partial\vec{v}_b} = 0$$

$$\frac{\partial y_g}{\partial\vec{x}_b} = \frac{\Delta\vec{x}_{1b}}{\rho_{1b}} - \frac{\Delta\vec{x}_{2b}}{\rho_{2b}}, \qquad \frac{\partial y_g}{\partial\vec{v}_b} = 0$$

Observation partial derivative w.r.t. to time

$$\frac{\partial y_g}{\partial t} = \dot{\rho}_{1a} - \dot{\rho}_{1b} - \left(\dot{\rho}_{2a} - \dot{\rho}_{2b}\right)$$

Observation time tag correlation

All ranges (ρ) are computed using the correction in the *Station-to-Satellite and Satellite-to-Satellite* ***1-way range*** algorithms described above.

Computed observation

$$y_c = y_g$$

3.17 Azimuth and Elevation from a Ground Station

Geometric observation

$$\Delta\vec{x} = \vec{x}_{sat} - \vec{x}_{sta}\,, \qquad \Delta\vec{v} = \vec{v}_{sat} - \vec{v}_{sta}$$

$$\rho = \left|\Delta\vec{x}\right|, \qquad \dot{\rho} = \frac{\Delta\vec{v}\cdot\Delta\vec{x}}{\rho}$$

Azimuth: $\cos\alpha = -\vec{e}\cdot\frac{\vec{u}\times\Delta\vec{x}}{\rho}$, $\sin\alpha = \vec{n}\cdot\frac{\vec{u}\times\Delta\vec{x}}{\rho}$. $\quad (y_g = \alpha)$

Elevation: $\sin\varepsilon = \frac{\vec{u}\cdot\Delta\vec{x}}{\rho}$. $\quad (y_g = \varepsilon)$

Observation partial derivatives w.r.t. satellite and station position and velocity

$$\frac{\partial\alpha}{\partial\vec{x}_{sat}} = -T^{-1}\frac{\vec{u}\times\Delta\vec{x}}{(\rho\cos\varepsilon)^2}\,, \qquad \frac{\partial\alpha}{\partial\vec{v}_{sat}} = 0$$

$$\frac{\partial\alpha}{\partial\vec{x}_{sta}} = ET^{-1}\frac{\vec{u}\times\Delta\vec{x}}{(\rho\cos\varepsilon)^2}\,, \qquad \frac{\partial\alpha}{\partial\vec{v}_{sta}} = 0$$

$$\frac{\partial\varepsilon}{\partial\vec{x}_{sat}} = \frac{\rho T^{-1}\vec{u} - \Delta\vec{x}\sin\varepsilon}{\rho^2\cos\varepsilon}\,, \qquad \frac{\partial\varepsilon}{\partial\vec{v}_{sat}} = 0$$

$$\frac{\partial\varepsilon}{\partial\vec{x}_{sta}} = -E\frac{\rho T^{-1}\vec{u} - \Delta\vec{x}\sin\varepsilon}{\rho^2\cos\varepsilon}\,, \qquad \frac{\partial\varepsilon}{\partial\vec{v}_{sta}} = 0$$

Observation partial derivative w.r.t. to time

$$\frac{\partial \alpha}{\partial t} = \frac{1}{\rho \sin \alpha}\left[\left(\vec{u} \times \Delta \vec{v} + \dot{\vec{u}} \times \Delta \vec{x} - \frac{\vec{u} \times \Delta \vec{x}}{\rho} \dot{\rho}\right) \cdot \vec{e} + \vec{u} \times \Delta \vec{x} \cdot \dot{\vec{e}}\right]$$

$$\frac{\partial \varepsilon}{\partial t} = \frac{1}{\cos \varepsilon}\left[\left(\frac{\Delta \vec{v}}{\rho} - \frac{\Delta \vec{x}}{\rho^2} \dot{\rho}\right) \cdot \vec{u} + \frac{\Delta \vec{x}}{\rho} \cdot \dot{\vec{u}}\right]$$

Observation time tag correlation

$$\Delta t_{ref} = \Psi \frac{\rho}{c}$$

Computed observation

$$\alpha_c = \alpha + \left(\Delta t_{tag} + \Delta t_{ref}\right) \frac{\partial \alpha}{\partial t} + \Delta \alpha$$

$$\varepsilon_c = \varepsilon + \left(\Delta t_{tag} + \Delta t_{ref}\right) \frac{\partial \varepsilon}{\partial t} + \Delta \varepsilon$$

4 ORBIT PROPAGATION

4.1 The Orbit Propagation Problem

The orbit propagation problem consists of predicting the satellite state vector (i.e. position and velocity of its centre-of-mass) in a future time t, given an initial state vector at time t_0 (the epoch) and a model for the forces acting on the satellite.

Therefore, propagating an orbit consists in the integration of the equations of motion. These are a vectorial second order set of differential equations, which can be converted to a first order set by taking into account that the velocity vector is the time derivative of the position vector.

$$\left.\begin{array}{l} \dfrac{d^2\vec{x}}{dt^2}=\vec{f}\left(t,\vec{x},\dfrac{d\vec{x}}{dt}\right) \\ \vec{x}(t=t_0)=\vec{x}_0 \\ \dfrac{d\vec{x}}{dt}(t=t_0)=\dot{\vec{x}}_0 \\ \vec{x}=(x,y,z) \\ \dot{\vec{x}}=\left(\dfrac{dx}{dt},\dfrac{dy}{dt},\dfrac{dz}{dt}\right) \end{array}\right\} \Rightarrow \frac{d\vec{x}}{dt}=\vec{v} \Rightarrow \left\{\begin{array}{l} \dfrac{d\vec{X}}{dt}=\vec{F}\left(t,\vec{X}\right) \\ \vec{X}(t=t_0)=\vec{X}_0 \\ \vec{X}=\left(x,y,z,v_x,v_y,v_z\right) \\ \vec{F}=\left(v_x,v_y,v_z,f_x,f_y,f_z\right) \end{array}\right. \qquad \text{Eq. 4.1}$$

Then, the final problem can be written as

$$\begin{array}{l} \dfrac{d\vec{X}}{dt}=\vec{F}\left(t,\vec{X}\right) \\ \vec{X}(t_0)=\vec{X}_0 \end{array}$$

where $\vec{X}$ may include other parameters than just position-velocity, called then extended state vector.

The problem then reduces to solve

$$\begin{array}{l} \dfrac{dy}{dx}=f(x,y) \\ y(x_0)=y_0 \end{array}$$

where in general *x, y* and *f* are vectorial functions.

A mathematical description of each of the forces acting on a near-earth satellite (gravitational, surface and propulsion forces) can be found in RD-5, RD-9, and RD-29. It is important to note that some of these force models include parameters whose numerical values are only imperfectly known.

4.2 Adams-Bashforth

This subroutine implements an 8th-order Adams-Bashforth/Adams-Moulton prediction-correction method to numerically solve a system of n ordinary first order differential equations with initial values RD-2. The problem can be expressed in the form:

$$\begin{aligned}
\frac{dy_1}{dx} &= f_1(x, y_1, y_2, \ldots, y_n) \quad ; \quad y_1(x_0) = y_0^1 \\
\frac{dy_2}{dx} &= f_2(x, y_1, y_2, \ldots, y_n) \quad ; \quad y_2(x_0) = y_0^2 \\
&\ldots \\
\frac{dy_n}{dx} &= f_n(x, y_1, y_2, \ldots, y_n) \quad ; \quad y_n(x_0) = y_0^n
\end{aligned}$$

being x the independent variable and y_i the dependent variables. Numerical methods evaluate the values of y_i in successive points of an interval starting at x_0 (*mesh points*). The solution at other intermediate points can be evaluated using an interpolation algorithm.

The system can be solved by solving each of the n equations simultaneously, so the problem is reduced to solving one differential equation previously indicated:

$$\frac{dy}{dx} = f(x, y, \ldots) \quad ; \quad y(x_0) = y_0$$

Multi-step methods evaluate y using information in several previous points of the interval. A particular case of these methods are the *predictor-corrector methods*. In a predictor-corrector method of order 8, y_{i+1} is predicted by a formula making use of the previous 9 points. This prediction is then used together with the previous 8 points in a formula that corrects the predicted value. The corrector formula is re-applied until it converges to a value of y_{i+1}. The convergence can be achieved in one or two iterations by selecting a small enough step size. Before applying a predictor-corrector method, the first 9 points must be available. These can be computed using a 8th-order single-step Runge-Kutta method (subroutine *rukut8*), starting from the known value of y at x_0, y_0.

The 8th-order Adams-Bashforth/Adams-Moulton method is based on the following algorithm:

$$\begin{aligned}
y_{n+1}^* &= y_n + \frac{h}{3628800} \sum_{i=0}^{8} \beta_i \cdot f_{n-i} \\
y_{n+1} &= y_n + \frac{h}{3628800} \left(\beta_0^* \cdot f\left(x_{n+1}, y_{n+1}^*\right) + \sum_{i=1}^{8} \beta_i^* \cdot f_{n-i+1} \right) \\
x_{n+1} &= x_n + h
\end{aligned}$$

the Near-Earth orbit propagation is studied. Some recommendations are also given about the optimal way to use these methods and some special actions that have to be taken in presence of force discontinuities. An eighth-order Adams-Bashforth/Adams-Moulton predictor-corrector method (initialised by an eighth-order Runge-Kutta method) is recommended. A description of this method can be found in [R1.].

The coefficients being (two of them are wrong in reference):

i	Adams-Bashforth Coefficients (β_i)	Adams-Moulton Coefficients (β_i^*)
0	14097247	1070017
1	-43125206	4467094
2	95476786	-4604594
3	-139855262	5595358
4	137968480	-5033120
5	-91172642	3146338
6	38833486	-1291214
7	-9664106	312874
8	1070017	-33953

Table 4.1: Adams-Bashford/Adams-Moulton Coefficients

4.3 Fixed step 8th order Runge-Kutta

This subroutine numerically integrates a system of ordinary differential equations using a single-step 8th-order Runge-Kutta method [R4.].

The formulation to compute the consecutive values of y_n at x_n, at independent variable increments *h,* by means of an 8th order single step Runge-Kutta method is

$$y_{n+1} = y_n + h\sum_{i=0}^{9} c_i \cdot k_i$$

$$x_{n+1} = x_n + h$$

$$k_0 = f(x_n, y_n)$$

$$k_i = f\left(x_n + h \cdot a_i, y_n + h\sum_{j=0}^{i-1} b_{ij} \cdot k_j \right) \quad ; \quad 1 \le i \le 9$$

where the required coefficients are listed in the following table

	j	0	1	2	3	4	5	6	7	8	9
i	**a_i**	**b_{ij}**									
1	4/27	4/27									
2	2/9	1/18	3/18								
3	1/3	1/12	0	3/12							

4	1/2	1/8	0	0	3/8						
5	2/3	13/54	0	-27/54	42/54	8/54					
6	1/6	389/4320	0	-54/4320	966/4320	-824/4320	243/4320				
7	1	-231/20	0	81/20	-1164/20	656/20	-122/20	800/20			
8	5/6	-127/288	0	18/288	-678/288	456/288	-9/288	576/288	4/288		
9	1	1481/820	0	-81/820	7104/820	-3376/820	72/820	-6.14634	-0.07317	720/820	
		41/840	0	0	27/840	272/840	27/840	216/840	0	216/840	41/840
		c_i									

Table 4.2: Runge-Kutta Coefficients

4.4 Variable step 7-8th order Runge-Kutta

The technique is to compute two Runge-Kutta methods of consecutive orders (7 and 8) at each step, RD-3. The lower order Runge-Kutta method provides an estimate for y_{n+1} while the higher one provides a more accurate estimate. Their difference is an estimate of the local truncation error for y_{n+1}:

$$TE = \left| y_{n+1} - y*_{n+1} \right| = A \cdot h^8 \text{ where } \left| A \right| \text{ is some coefficient}$$

$$\hat{y}_{n+1} = y_n + h \sum_{i=0}^{10} c_i \cdot k_i$$

$$y_{n+1} = y_n + h \sum_{i=0}^{12} c*_i \cdot k_i$$

$$k_0 = f(x, y_n)$$

$$k_i = f\left(x + h \cdot a_i, y_n + h \sum_{j=0}^{i-1} b_{ij} \cdot k_j \right) \quad ; \quad 1 \le i \le 12$$

$$TE = \left| \hat{y}_{n+1} - y_{n+1} \right| = \frac{41}{840} \cdot h \cdot \left| k_0 + k_{10} - k_{11} - k_{12} \right|$$

$$A = \sum_{components} \left(\frac{TE_j}{tol_j} \right)^2 < 1 \quad ; \quad (convergence)$$

$$h_{new} = h \cdot \left(\frac{0.0025}{A} \right)^{\frac{1}{16}}$$

	j	0	1	2	3	4	5	6	7	8	9	10	11		
i	a_i	b_{ij}												c_i	c_i^*
0	2/27	0												41/840	0
1	1/9	2/27												0	
2	1/6	1/36	1/12											0	
3	5/12	1/24	0	1/8										0	
4	1/2	5/12	0	-25/16	25/16									0	
5	5/6	1/20	0	0	1/4	1/5								34/105	
6	1/6	-25/108	0	0	125/108	-65/27	125/54							9/35	
7	2/3	31/300	0	0	0	61/225	-2/9	13/900						9/35	
8	1/3	2	0	0	-53/6	704/45	-107/9	67/90	3					9/280	
9	1	-91/108	0	0	23/108	-976/135	311/54	-19/60	17/6	-1/12				9/280	
10	0	2383/4100	0	0	-341/164	4496/1025	-301/82	2133/4100	45/82	45/164	18/41			41/840	0
11	1	3/205	0	0	0	0	-6/41	-3/205	-3/41	3/41	6/41	0		0	41/840
12	2/27	-1777/4100	0	0	-341/164	4496/1025	-289/82	2193/4100	51/82	33/164	12/41	0	1	0	41/840

Table 4.3: Runge-Kutta 7-8 Coefficients

4.5 Legendre Polynomials and Associated Functions

As it has been seen, a set of Legendre functions has to be defined to calculate some of the perturbing forces. These functions are also needed for site displacement.For that reason a dedicated module has been written to handle with these computation within NAPEOS.

4.5.1 Definition

The Legendre Function of the first kind are solutions to the case of null parameter Legendre Differential Equation (RD-16) which is obtained from the assumption of a separation of variables in the Laplace equation written in spherical coordinates (RD-6) Applying the Rodriges' Formula (RD-16), one obtains the definition of the Legendre polynomials as:

$$P_n(x) = \frac{1}{2^n n!}\frac{d^n}{dx^n}(x^2-1)^n$$

The polynomials can be evaluated using the recurrence relationship below

$$P_n(x) = \frac{1}{n}\left[(2n-1)xP_{n-1}(x) - (n-1)P_{n-2}(x)\right]$$

with

$$P_0(x) = 1$$

$$P_1(x) = x$$

The associated Legendre Functions are solution of the associated Legendre differential equation (parameter not null), and can be defined as

$$P_n^m(x) = (1-x^2)^{\frac{m}{2}} \frac{d^m}{dx^m} P_n(x) \quad , \quad n \geq m$$

These functions can also be evaluated using two recurrence relations

$$\begin{cases} P_n^m(x) = \dfrac{1}{n}\left[(2n-1)\, x\, P_{n-1}^m(x) - (n+m-1)\, P_{n-2}^m(x)\right] & , \quad n-2 > m \\ P_n^m(x) = \dfrac{1}{n}\left[(2n-1)\, x\, P_{n-1}^m(x)\right] & , \quad n-2 \leq m \end{cases}$$

$$\begin{cases} P_n^m(x) = (2n-1)\sqrt{1-x^2}\, P_{n-1}^{m-1}(x) + P_{n-2}^m(x) & , \quad n-2 > m \\ P_n^m(x) = (2n-1)\sqrt{1-x^2}\, P_{n-1}^{m-1}(x) & , \quad n-2 \leq m \end{cases}$$

For some calculations these functions can be defined in an alternative form, in order to save computation time. For instance, another set of functions can be formed using the following relationship

$$Q_n^m(x) = \frac{1}{(1-x^2)^{m/2}\, m!} P_n^m(x)$$

These functions can be obtained using recurrences, in a similar way as for Legendre functions:

$$\begin{cases} Q_n^m(x) = \dfrac{2n-1}{m} Q_{n-1}^{m-1}(x) + Q_{n-2}^m(x) & , \quad n-2 > m \\ Q_n^m(x) = \dfrac{2n-1}{m} Q_{n-1}^{m-1}(x) & , \quad n-2 \leq m \end{cases}$$

A mean advantage of this set is that derivatives w.r.t. its argument are much easier to be evaluated as shown below:

$$\frac{d}{dx} Q_n^m(x) = (m+1) Q_n^{m+1}(x)$$

5 ORBIT FORCE MODELS

5.1 Central Earth Gravity Field

This is the main driving force acting on a satellite, which is equivalent to the field generated by a mass point located at the centre of co-ordinates. All other forces are considered as perturbation to first order expansion about this one. The acceleration generated by such a field at a point define by its position vector, $\vec{r}$, is given by

$$\vec{a} = \frac{\mu_{\oplus}}{r^3}\vec{r}$$

or written as a potential function

$$U = \frac{\mu_{\oplus}}{r}$$

where

$\mu_{\oplus} = GM_{\oplus}$: Gravitational constant of the Earth

The partial derivatives of the central acceleration with respect to the satellite position are then

$$\frac{\partial a_i}{\partial x_j} = \frac{\partial^2 \ddot{x}_i}{\partial x_j{}^2} = 3\frac{\mu_{\oplus}}{r^5}x_i x_j \quad , \ i \neq j$$

$$\frac{\partial a_i}{\partial x_j} = \frac{\partial^2 \ddot{x}_i}{\partial x_j{}^2} = 3\frac{\mu_{\oplus}}{r^5}x_i^2 - \frac{\mu_{\oplus}}{r^3} \ , \ i = j$$

To calculate the partial w.r.t. to $\mu_{\oplus}$, the perturbing acceleration due to other effects, like non-central earth gravity field, must be already computed. Since the dependency on $\mu_{\oplus}$ is linear, the derivative is simple the acceleration due to earth gravity divided by $\mu_{\oplus}$.

5.2 Non-spherical Earth Gravity Field

The non-spherical mass distribution of the Earth creates deviations from the central Earth gravity field. This deviation is considered as a correction to the central field and is normally expressed as a spherical harmonic expansion in the form

$$\Delta U = \frac{\mu_{\oplus}}{r}\left[\sum_{n=2}^{\infty} C_{n0}\left(\frac{R_{\oplus}}{r}\right)^n P_n(\sin\phi) + \sum_{n=2}^{\infty}\left(\frac{R_{\oplus}}{r}\right)^n \sum_{m=1}^{n} P_{nm}(\sin\phi)\left[C_{nm}\cos m\lambda + S_{nm}\sin m\lambda\right]\right] \qquad \text{Eq. 5.1}$$

where

r : radius vector

ϕ : latitude

λ : longitude

P_n : Legendre polynomial of degree n

P_{nm} : Legendre associated function of degree n and order m

$R_\oplus$: Radius of the Earth (reference for the expansion)

C_{n0} : zonal harmonic coefficients

C_{nm}, S_{nm} : tesseral ($n{\neq}m$) harmonic coefficients and sectoral harmonic coefficients ($n{=}m$)

There are several models of the Geopotential published by the *Group de Recherche en Géodésie Spatiale*, GeoForschungsZentrum, the Centre for Space Research and the Goddard Space Flight Centre for instance.

5.2.1 Acceleration Computation Algorithm

The calculation of the gradient of this potential function is quite time-consuming. Therefore an optimized algorithm must be used to save time in repetitive calculation. The equation is transformed first taking into account the following definition

$$\rho = \sqrt{x^2 + y^2} = r\cos\phi$$

$$c_m = \rho^m \cos m\lambda = r^m \cos^m\phi \cos m\lambda$$

$$s_m = \rho^m \sin m\lambda = r^m \cos^m\phi \sin m\lambda$$

c_m and s_m are given by recurrence relationship as

$$c_m = c_1 c_{m-1} - s_1 s_{m-1}$$

$$s_m = c_1 s_{m-1} - s_1 c_{m-1}$$

with

$$c_1 = \rho\cos\lambda$$

$$s_1 = \rho\sin\lambda$$

A set of functions derived from Legendre polynomials (Section 4.5) is defined as below:

$$Q_{nm}(x) = \frac{1}{m!}\frac{d^m}{dx^m}P_n(x)$$

These functions are generated by **OPlg_legn** in module **OPlg_Legendre**. The derivatives are also expressed in terms of functions of the same set

$$\frac{d}{dx}Q_{nm}(x) = (m+1)Q_{n(m+1)}(x)$$

Another improvement on time saving is the use of factorised harmonic coefficient as follows

$$C'_{nm} = m!\; C_{nm}$$

$$S'_{nm} = m!\; S_{nm}$$

$$\Delta U = \frac{\mu_\oplus}{r}\sum_{n=2}^{\infty}\left(\frac{R_\oplus}{r}\right)^n \sum_{m=0}^{n}\frac{Q_{nm}(\sin\phi)}{r^m}\left[C'_{nm}c_m + S'_{nm}s_m\right] \qquad \text{Eq. 5.2}$$

The acceleration in earth-fixed system are calculated as follow

$$\vec{a} = \ddot{\vec{x}} = \nabla(\Delta U) = \begin{cases} \ddot{x} = -\frac{\mu_\oplus}{r^3}\left(\sigma_1\, x\, r + \sigma_2\, x\, z - \sigma_3\, r^2\right) \\ \ddot{y} = -\frac{\mu_\oplus}{r^3}\left(\sigma_1\, y\, r + \sigma_2\, y\, z - \sigma_4\, r^2\right) \\ \ddot{z} = -\frac{\mu_\oplus}{r^3}\left(\sigma_1\, z\, r - \sigma_2\, r^2 \cos^2\phi\right) \end{cases}$$

where

$$\sigma_1 = \sum_{n=2}^{\infty}\left(\frac{R_\oplus}{r}\right)^n \sum_{m=0}^{n}(n+m+1)\frac{Q_{nm}(\sin\phi)}{r^m}\left[C'_{nm}c_m + S'_{nm}s_m\right]$$

$$\sigma_2 = \sum_{n=2}^{\infty}\left(\frac{R_\oplus}{r}\right)^n \sum_{m=0}^{n}(m+1)\frac{Q_{n(m+1)}(\sin\phi)}{r^m}\left[C'_{nm}c_m + S'_{nm}s_m\right]$$

$$\sigma_3 = \sum_{n=2}^{\infty}\left(\frac{R_\oplus}{r}\right)^n \sum_{m=0}^{n}m(m+1)\frac{Q_{n(m+1)}(\sin\phi)}{r^{m-1}}\left[C'_{nm}c_{m-1} + S'_{nm}s_{m-1}\right]$$

$$\sigma_4 = \sum_{n=2}^{\infty}\left(\frac{R_\oplus}{r}\right)^n \sum_{m=0}^{n}m(m+1)\frac{Q_{n(m+1)}(\sin\phi)}{r^{m-1}}\left[S'_{nm}c_{m-1} + C'_{nm}s_{m-1}\right]$$

After this computation, the acceleration vector must be transformed to the inertial frame in order to add it to the accumulated perturbing acceleration.

5.2.2 Partial Derivatives Computation Algorithm

For orbit determination, the partial derivatives of the acceleration with respect to the state vector are also required. The acceleration depends only on the spacecraft position and therefore the partial derivatives with respect to velocity are all zero.

Taking into account the relationship

$$\frac{\partial^2 U}{\partial x_i \partial x_j} = \frac{\partial^2 U}{\partial x_j \partial x_i}$$

the partials are given as a matrix like this

$$\frac{\partial \vec{a}}{\partial \vec{x}} = \begin{bmatrix} \frac{\partial a_x}{\partial x} & \frac{\partial a_y}{\partial x} & \frac{\partial a_z}{\partial x} \\ \frac{\partial a_x}{\partial y} & \frac{\partial a_y}{\partial y} & \frac{\partial a_z}{\partial y} \\ \frac{\partial a_x}{\partial z} & \frac{\partial a_y}{\partial z} & \frac{\partial a_z}{\partial z} \end{bmatrix} = \begin{bmatrix} \frac{\partial^2 U}{\partial x^2} & \frac{\partial^2 U}{\partial y \partial x} & \frac{\partial^2 U}{\partial z \partial x} \\ \frac{\partial^2 U}{\partial x \partial y} & \frac{\partial^2 U}{\partial y^2} & \frac{\partial^2 U}{\partial y \partial z} \\ \frac{\partial^2 U}{\partial x \partial z} & \frac{\partial^2 U}{\partial z \partial y} & \frac{\partial^2 U}{\partial z^2} \end{bmatrix} = \begin{bmatrix} \frac{\partial^2 U}{\partial x^2} & \frac{\partial^2 U}{\partial y \partial x} & \frac{\partial^2 U}{\partial z \partial x} \\ \frac{\partial^2 U}{\partial y \partial x} & \frac{\partial^2 U}{\partial y^2} & \frac{\partial^2 U}{\partial y \partial z} \\ \frac{\partial^2 U}{\partial z \partial x} & \frac{\partial^2 U}{\partial y \partial z} & \frac{\partial^2 U}{\partial z^2} \end{bmatrix} = \begin{bmatrix} \ddot{x}'_x & \ddot{x}'_y & \ddot{x}'_z \\ \ddot{x}'_y & \ddot{y}'_y & \ddot{y}'_z \\ \ddot{x}'_z & \ddot{y}'_z & \ddot{z}'_z \end{bmatrix}$$

the six unique partial derivatives can the n be obtained from the following formulation

$$\begin{bmatrix} \ddot{x}'_x \\ \ddot{x}'_y \\ \ddot{x}'_z \\ \ddot{y}'_y \\ \ddot{y}'_z \\ \ddot{z}'_z \end{bmatrix} = -\frac{\mu_\oplus}{r^3} \begin{bmatrix} -\bar{x}^2 & 1 & -\bar{x}^2 & 2\bar{x} & -\bar{x}^2\bar{z}^2 & \bar{z} & -\bar{x}^2\bar{z} & 2\overline{xz} & -1 & 0 & 0 & 0 & 0 \\ -\overline{xyz} & 0 & -\overline{xy} & \bar{y} & -\overline{xyz}^2 & 0 & -\overline{xyz} & \overline{yz} & 0 & \bar{x} & \overline{xz} & -1 & 0 \\ \bar{\rho}^2\bar{x} & 0 & -\overline{xz} & \bar{z} & \bar{\rho}^2\overline{xz} & \bar{x} & -\overline{xz}^2 & -\bar{\rho}^2 & 0 & 0 & 0 & 0 & 0 \\ -\bar{y}^2\bar{z} & 1 & -\bar{y}^2 & 0 & -\bar{y}^2\bar{z}^2 & \bar{z} & -\bar{y}^2\bar{z} & 0 & 0 & 2\bar{y} & 2\overline{yz} & 0 & -1 \\ \bar{\rho}^2\bar{y} & 0 & -\overline{yz} & 0 & \bar{\rho}^2\overline{yz} & \bar{y} & -\overline{yz}^2 & 0 & 0 & \bar{z} & -\bar{\rho}^2 & 0 & 0 \\ -\bar{\rho}^2\bar{z} & 1 & -\bar{z}^2 & 0 & -\bar{\rho}^2 & 0 & -\bar{\rho}^2\bar{z} & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} \sigma_1 \\ \sigma_2 \\ \sigma_3 \\ \sigma_4 \\ \sigma_5 \\ \sigma_6 \\ \sigma_7 \\ \sigma_8 \\ \sigma_9 \\ \sigma_{10} \\ \sigma_{11} \\ \sigma_{12} \\ \sigma_{13} \end{bmatrix}$$

where

$$\bar{x} = \frac{x}{r} \quad , \quad \bar{y} = \frac{y}{r} \quad , \quad \bar{z} = \frac{z}{r} \quad , \quad \bar{\rho} = \frac{\rho}{r} = \frac{\sqrt{x^2 + y^2}}{r} = \sqrt{\bar{x}^2 + \bar{y}^2}$$

and

$$\sigma_1 = \sum_{n=2}^{N} \left(\frac{R_\oplus}{r} \right)^n \sum_{m=0}^{\min(n-1,M-1)} (n+m+1)(m+1) \frac{Q_{n(m+1)}(\sin\phi)}{r^m} \left[C'_{nm} c_m + S'_{nm} s_m \right]$$

$$\sigma_2 = \sum_{n=2}^{N} \left(\frac{R_\oplus}{r} \right)^n \sum_{m=0}^{\min(n,M)} (n+m+1) \frac{Q_{nm}(\sin\phi)}{r^m} \left[C'_{nm} c_m + S'_{nm} s_m \right]$$

$$\sigma_3 = \sum_{n=2}^{N} \left(\frac{R_\oplus}{r} \right)^n \sum_{m=0}^{\min(n,M)} (n+m+1)(n+m+3) \frac{Q_{nm}(\sin\phi)}{r^m} \left[C'_{nm} c_m + S'_{nm} s_m \right]$$

$$\sigma_4 = \sum_{n=2}^{N} \left(\frac{R_\oplus}{r} \right)^n \sum_{m=1}^{\min(n,M)} m(n+m+1) \frac{Q_{nm}(\sin\phi)}{r^{m-1}} \left[C'_{nm} c_{m-1} + S'_{nm} s_{m-1} \right]$$

$$\sigma_5 = \sum_{n=2}^{N} \left(\frac{R_\oplus}{r} \right)^n \sum_{m=0}^{\min(n-2,M-2)} (m+2) \frac{Q_{n(m+2)}(\sin\phi)}{r^{m-1}} \left[C'_{nm} c_{m-1} + S'_{nm} s_{m-1} \right]$$

$$\sigma_6 = \sum_{n=2}^{N} \left(\frac{R_\oplus}{r} \right)^n \sum_{m=0}^{\min(n-1,M-1)} (m+1) \frac{Q_{n(m+1)}(\sin\phi)}{r^m} \left[C'_{nm} c_m + S'_{nm} s_m \right]$$

$$\sigma_7 = \sum_{n=2}^{N} \left(\frac{R_\oplus}{r} \right)^n \sum_{m=0}^{\min(n-1,M-1)} (n+m+4)(m+1) \frac{Q_{n(m+1)}(\sin\phi)}{r^m} \left[C'_{nm} c_m + S'_{nm} s_m \right]$$

$$\sigma_8 = \sum_{n=2}^{N} \left(\frac{R_\oplus}{r} \right)^n \sum_{m=1}^{\min(n,M)} m(m+1) \frac{Q_{n(m+1)}(\sin\phi)}{r^{m-1}} \left[C'_{nm} c_{m-1} + S'_{nm} s_{m-1} \right]$$

$$\sigma_9 = \sum_{n=2}^{N} \left(\frac{R_\oplus}{r} \right)^n \sum_{m=2}^{\min(n,M)} m(m-1) \frac{Q_{nm}(\sin\phi)}{r^{m-2}} \left[C'_{nm} c_{m-2} + S'_{nm} s_{m-2} \right]$$

$$\sigma_{10} = \sum_{n=2}^{N} \left(\frac{R_\oplus}{r} \right)^n \sum_{m=1}^{\min(n,M)} m(n+m+1) \frac{Q_{nm}(\sin\phi)}{r^{m-1}} \left[S'_{nm} c_{m-1} - C'_{nm} s_{m-1} \right]$$

$$\sigma_{11} = \sum_{n=2}^{N} \left(\frac{R_\oplus}{r} \right)^n \sum_{m=1}^{\min(n,M)} m(m+1) \frac{Q_{n(m+1)}(\sin\phi)}{r^{m-1}} \left[S'_{nm} c_{m-1} - C'_{nm} s_{m-1} \right]$$

$$\sigma_{12} = \sum_{n=2}^{N} \left(\frac{R_\oplus}{r} \right)^n \sum_{m=2}^{\min(n,M)} m(m-1) \frac{Q_{nm}(\sin\phi)}{r^{m-2}} \left[S'_{nm} c_{m-2} - C'_{nm} s_{m-2} \right]$$

$$\sigma_{13} = \sum_{n=2}^{N} \left(\frac{R_{\oplus}}{r} \right)^{n} \sum_{m=2}^{\min(n,M)} m(m-1) \frac{Q_{nm}(\sin\phi)}{r^{m-2}} \left[-C'_{nm} c_{m-2} - S'_{nm} s_{m-2} \right]$$

with

N : truncation degree in the partial derivative expansion

M : truncation order in the partial derivative expansion

5.3 Secular rates to harmonics

Some temporal variations in the geopotential can be due to internal geophysical phenomena that are difficult to model and which lead, on the flight dynamics scale, to a temporal drift of the harmonic coefficients.

$$C_{nm} = C_{nm}^{0} + C_{nm}^{1}(t - t_0)$$
$$S_{nm} = S_{nm}^{0} + S_{nm}^{1}(t - t_0)$$

where

C_{nm}^{0} and S_{nm}^{0} are the value of the harmonic for a reference epoch (for instance, 1st January 1986, for JGM-1)

C_{nm}^{1} and S_{nm}^{1} are the secular rates for the harmonic (in $year^{-1}$ usually)

The change in $C_{20} = J_2$ due to the deceleration of the Earth rotation is also considered.

Secular change in the harmonics C_{21} and S_{21} due to the displacement of the pole inertia are also considered, section 6 of RD-7 which gives

$$\overline{C_{21}} = +\sqrt{3}\,\overline{x}\,\overline{C_{20}} - \overline{x}\,\overline{C_{22}} + \overline{y}\,\overline{S_{22}}$$
$$\overline{S_{21}} = -\sqrt{3}\,\overline{y}\,\overline{C_{20}} - \overline{y}\,\overline{C_{22}} - \overline{x}\,\overline{S_{22}}$$

Where x and y are the angular displacements of the pole of the Terrestrial Reference Frame.

5.4 Solid Earth Tides Perturbation

The earth, which is not perfectly rigid, deforms under the solar and lunar gravitational attractions. These deformations, associated with the redistribution of mass, can be conveniently defined using the Love numbers, introduced by A. love in 1909. The deformation can be expressed as a change to the external geo-potential, see section 6 of RD-7

$$\Delta U = \frac{\mu_\oplus}{r}\sum_{l=2}^{\infty}\sum_{k=1}^{l}\left(\frac{R_\oplus}{r}\right)^{l} P_{lk}(\sin\phi)\left[\Delta C_{lk}\cos m\lambda + \Delta S_{lk}\sin m\lambda\right]$$

with

$$\Delta C_{nk} = k_n\left(2-\delta_{0k}\right)\frac{(n-k)!}{(n+k)!}\sum_{j=1,2}\frac{M_j}{M_T}\left(\frac{R_\oplus}{r_j}\right)P_{nk}\left(\sin\phi_j\right)\cos k\left(\lambda_j-\phi\right)$$

$$\Delta S_{nk} = k_n\left(2-\delta_{0k}\right)\frac{(n-k)!}{(n+k)!}\sum_{j=1,2}\frac{M_j}{M_T}\left(\frac{R_\oplus}{r_j}\right)P_{nk}\left(\sin\phi_j\right)\sin k\left(\lambda_j-\phi\right)$$

where

k_n is the Love number for degree n

ϕ is the phase due to the inelasticity of the Earth

Considering only the second order effect, i.e. ΔC_{20}, a first simplification can be obtained. With this approximation, indirect effects on ΔC_{4k} and ΔS_{4k} are neglected.

$$\Delta U = k_2\sum_{P=1}^{2}\frac{\mu_P R_\oplus{}^5}{2r_P{}^3 r^3}\left[3\frac{\left(\vec{r}\cdot\vec{r}_P\right)^2}{r^2 r_P{}^2}-1\right]$$

where

k_2 is the Love number for degree 2

$\mathrm{P}=1,2$ corresponds to Moon and Sun

$\vec{r}$ is the s/c geocentric position vector

$\vec{r}_j$ is the s/c geocentric position vector of body j, with a phase ϕ, to account for inelasticity.

The gradient of this perturbating potential is written then

$$\Delta\vec{a} = k_2\sum_{P=1}^{2}\frac{\mu_P R_\oplus{}^5}{2r_P{}^3 r^5}\left[-\frac{-15\left(\vec{r}\cdot\vec{r}_P\right)^2}{r^2 r_P{}^2}\vec{r}+3\vec{r}+\frac{6\left(\vec{r}\cdot\vec{r}_P\right)}{r_P}\vec{r}_P\right]$$

In the deformation of the Earth, a time lag has to be considered due to inelasticity of the solid part of the Earth. This phase angle is approximately 2.9 degrees, so the global tide occurs at points that were directly in line with the perturbing body (Moon or Sun) 12 minutes ago RD-6.

The acceleration partials are evaluated as

$$\frac{\partial \ddot{r}_i}{\partial r_j} = \frac{k_2 \mu_P R_\oplus{}^5}{2 r_P{}^3 r^5} \left[-\frac{30(\vec{r}\cdot\vec{r}_P)^2}{r^2 r_P{}^2} r_i r_{Pj} + \frac{105(\vec{r}\cdot\vec{r}_P)}{r^4 r_P{}^2} r_i r_j - \frac{15}{r^2} r_i r_j + \frac{6}{r_P{}^2} r_{Pi} r_{Pj} - \frac{30(\vec{r}\cdot\vec{r}_P)}{r^2 r_P{}^2} r_j r_{Pi} \right] \quad (i \neq j)$$

$$\frac{\partial \ddot{r}_i}{\partial r_i} = \frac{k_2 \mu_P R_\oplus{}^5}{2 r_P{}^3 r^5} \left[-\frac{60(\vec{r}\cdot\vec{r}_P)^2}{r^2 r_P{}^2} r_i r_{Pi} + \frac{105(\vec{r}\cdot\vec{r}_P)}{r^4 r_P{}^2} r_i{}^2 - \frac{15}{r^2} r_i{}^2 + \frac{6}{r_P{}^2} r_{Pi}{}^2 - \frac{15(\vec{r}\cdot\vec{r}_P)}{r^2 r_P{}^2} + 3 \right]$$

Some corrections to ΔC_{nk} and ΔS_{nk} must be taken into account due to frequency dependent tides; only for the harmonics coefficients 21 and 22 are considered. At this point only elastic Earth is considered.

$$\Delta C_{21} = A_1 \sum_{S(21)} \delta k_s H_s \sin\theta_s$$

$$\Delta S_{21} = A_1 \sum_{S(21)} \delta k_s H_s \cos\theta_s$$

$$\Delta C_{22} = A_2 \sum_{S(22)} \delta k_s H_s \cos\theta_s$$

$$\Delta S_{22} = -A_2 \sum_{S(22)} \delta k_s H_s \sin\theta_s$$

where

δk_s is the difference between Wahr model for k at frequency s and the nominal value k2.

H_s is amplitude of term at frequency s (see RD-7).

Note that in Napeos the solid earth tides according to the response of the “anelastic Earth” has been implemented according to the Section 6.1 of RD-7.

5.5 Pole Tides Perturbation

Another correction present in the formulae is the Earth Pole Tide, due to the centrifugal effect of polar motion. From RD-7 the formulae below has been adopted.

$$\Delta\overline{C_{21}} = -1.333\times10^{-9}(m_1 - 0.0115 m_2)$$

$$\Delta\overline{S_{21}} = -1.333\times10^{-9}(m_2 + 0.0115 m_1)$$

Where m_1 and m_2 are in seconds of arc and defined, according to Section 7.1 of RD-7, as:

$$m_1 = +(x_p - \overline{x}_p)$$

$$m_2 = -(y_p - \overline{y}_p)$$

where x_p, y_p are expressed in arc of seconds and

$$\bar{x}_p(t) = \bar{x}_p(t_0) + (t - t_0)\dot{\bar{x}}_p(t_0)$$

$$\bar{y}_p(t) = \bar{y}_p(t_0) + (t - t_0)\dot{\bar{y}}_p(t_0)$$

Using the given values (in arcseconds and arcseconds per year respectively):

$$\bar{x}_p(t_0) = 0.054 \qquad \dot{\bar{x}}_p(t_0) = 0.00083$$

$$\bar{y}_p(t_0) = 0.357 \qquad \dot{\bar{y}}_p(t_0) = 0.00395$$

These values are defined in the module DBcb_Data.f90.

5.6 Ocean Tides

The dynamical contribution of ocean tides due to the gravitational attraction of the Sun and Moon can be formulated in terms of a large number of frequency-dependent terms. The additional potential outside the Earth can be expressed as an expansion of spherical harmonics, see e.g., RD-7.

$$\Delta U = 4\pi G R_\oplus \rho_w \sum_{l=2}^{\infty}\sum_{m=0}^{l}\sum_{s}\frac{1+k_l'}{2nl+1}\left(\frac{R_\oplus}{r}\right)^{n+1}\left(A_{s,l,m}\cos m\lambda + B_{s,l,m}\sin m\lambda\right) \qquad \text{Eq. 5.3}$$

Where

ρ_w is the density of sea water

k_l' is load deformation coefficient

s is the tide index

$$A_{s,l,m} = \left(C_{s,l,m}^{+} + C_{s,l,m}^{-}\right)\cos\theta_s + \left(S_{s,l,m}^{+} + S_{s,l,m}^{-}\right)\sin\theta_s$$

$$B_{s,l,m} = \left(S_{s,l,m}^{+} - S_{s,l,m}^{-}\right)\cos\theta_s - \left(C_{s,l,m}^{+} - C_{s,l,m}^{-}\right)\sin\theta_s$$

where

$\theta_s = \vec{n}_s \cdot \vec{\beta}$ is the argument of the tide with $\vec{n}_s = \left[n_1\ n_2\ n_3\ n_4\ n_5\ n_6\right]_s^l$ which is the vector formed with the integer multipliers of the tide. These integers are related with **Doodson** notation $d_1\ d_2\ d_3\ d_4\ d_5\ d_6$ in the following way

$$d_1 = n_1$$

$$d_i = n_i + 5 \quad (i = 2,3,4,5,6)$$

The **Doodson** vector $\vec{\beta} = [\tau \; s \; h \; p \; N' \; p_s]^T$, is related to the fundamental arguments of nutation. For details, see RD-7, RD-8, and RD-9.

Eq. 5.3 shows that the effect of ocean tides can be translated into a correction to the gravitational spherical harmonic of Eq. 5.1.

$$\Delta C_l^m = \frac{4\pi R_\oplus^{\;2} \rho_w}{M_\oplus} \cdot \frac{1+k_l'}{2l+1} \sum_s A_{s,l,m}$$

$$\Delta S_l^m = \frac{4\pi R_\oplus^{\;2} \rho_w}{M_\oplus} \cdot \frac{1+k_l'}{2l+1} \sum_s B_{s,l,m}$$

where $M_\oplus$ is the Earth's mass.

The ocean tides effects are accounted for on user request. It requires an input file that specifies the amplitudes and the phases. The user has to indicate, just like for the gravity field, the degree and order for which the ocean tidal effect on the gravity has to be computed. In addition the user has to maximum number of constituents to be used. For this purpose the ocean tides constituents in the input file are sorted by size of their amplitudes so that only the largest “n” constituents are used.

5.7 Third Body gravitational force

The presence of the Moon, the Sun and the Planets produces acceleration on a spacecraft orbiting the Earth that responds to the following formulation

$$\Delta \vec{a} = \sum_{k=1}^{n} \mu_k \left[\frac{\vec{r}_k - \vec{r}}{\left|\vec{r}_k - \vec{r}\right|^3} - \frac{\vec{r}_k}{\left|\vec{r}_k\right|^3} \right]$$

where the *k*-summation extends to all perturbing bodies and μ_k is the gravitational constant of body *j*.

In NAPEOS all planets, including Pluto, are considered.

5.8 Indirect J_2/Moon Interaction

The third-body perturbation equation is obtained assuming that the gravitational attraction between the Earth and the rest of planets (including the Sun) can be modelled as point centred masses. This approximation can bring errors for the case of the Moon, since its proximity makes it be affected by the non-spherical mass distribution of the Earth.

A first approximation, only J_2 of the Earth harmonic geo-potential expansion is considered. In Earth-fixed system this perturbation is expressed as

$$\Delta\vec{a} = \frac{3}{2} J_2 \frac{\mu_l}{r_l^2} \left(\frac{R_\oplus}{r_l} \right)^2 \begin{bmatrix} \left(5\sin^2\phi_l - 1\right)\cos\phi_l \cos\lambda_l \\ \left(5\sin^2\phi_l - 1\right)\cos\phi_l \sin\lambda_l \\ \left(5\sin^2\phi_l - 3\right)\sin\phi_l \end{bmatrix}$$

where

ϕ_l is the latitude of the Moon

λ_l is the longitude of the Moon

μ_l is the gravity constant of the Moon

J_2 is the second zonal harmonic of the Earth gravity field

5.9 General Relativistic effects

The acceleration acting on a satellite orbiting a heavy rotating body must be corrected for relativistic effects due to the curvature in the time-space.

The correction to the acceleration of an artificial satellite by the presence of the Earth as given in RD-7 is:

$$\Delta\ddot{\vec{r}} = -\frac{\mu_\oplus}{c^2 r^3} \left\{ \left[2(\beta+\gamma)\frac{\mu_\oplus}{r} - \gamma\left(\dot{\vec{r}} \times \dot{\vec{r}}\right) \right] \vec{r} + 2(1+\gamma)(\vec{r}\cdot\vec{v})\vec{v} \right\}$$

$$+ (1+\gamma)\frac{\mu_\oplus}{c^2 r^3} \left[\frac{3}{r^2}\left(\vec{r} \times \dot{\vec{r}}\right)\left(\vec{r}\cdot\vec{J}\right) + \left(\dot{\vec{r}}\cdot\vec{J}\right) \right]$$

$$+ (1+2\gamma)\left[\dot{\vec{R}} \times \left(\frac{-\mu_{Sun}\vec{R}}{c^2 R^3} \right) \times \dot{\vec{r}} \right]$$

Where

c : speed of light

β, γ : PPN parameters, equal to 1 in General Relativity

$\vec{J}$ is the Earth's angular momentum per unit mass $\left(\left|\vec{J}\right| \cong 9.8\times10^8\ m^2/s\right)$

$\vec{r}, \dot{\vec{r}}, \ddot{\vec{r}}$: geocentric satellite position, velocity and acceleration, respectively

$\mu_\oplus, \mu_{Sun}$: Gravitational constant of the Earth, and Sun, respectively

The second and third lines in the equation above are the Lense-Thirring precession (frame-dragging) and the geodesic (de Sitter) precession, respectively. Both terms are currently neglected in NAPEOS. Only the first term is included. With both β and γ equal to one for general relativity the relativistic correction becomes:

$$\Delta\ddot{\vec{r}} = -\frac{\mu_\oplus}{c^2 r^3}\left\{\left[4\frac{\mu_\oplus}{r} - \left(\dot{\vec{r}} \times \dot{\vec{r}}\right)\right]\vec{r} + 4\left(\vec{r}\cdot\vec{v}\right)\vec{v}\right\}$$

This is implemented in the module OPre_Relativity.f90.

5.10 Solar Radiation Pressure

The perturbing forces due to solar radiation pressure may be computed using two different models.

5.10.1 IERS Solar Radiation Pressure Model

The IERS Formulation is given in RD-7

$$\Delta\vec{a} = K^* C_R \frac{\vec{R}}{R}$$

where

C_R : Solar radiation coefficients (scale factor)

$\vec{R}, R$: Heliocentric radius vector (and its module) of the satellite

$$K^* = \frac{I_{AU}}{c}\left(\frac{R_{AU}}{R}\right)^2 \frac{A_R}{m} \qquad \text{Eq. 5.4}$$

where

I_{AU} : Solar intensity of radiation at one astronomical unit

R_{AU} : One astronomical unit

A_R : Cross sectional area (as seen from the Sun)

m : Satellite mass

c: speed of light

Constant K^* (Eq. 5.4)is obtained and so the partial w.r.t. s/c state vector that can be written as

$$\frac{\partial^2 \ddot{r}_i}{\partial {r_j}^2} = -C_R \frac{A_R}{m} k \frac{{R_{AU}}^2}{R^3} \left[\frac{3R_i R_j}{R^2} \right] \quad (i \neq j)$$

$$\frac{\partial^2 \ddot{r}_i}{\partial {r_i}^2} = -C_R \frac{A_R}{m} k \frac{{R_{AU}}^2}{R^3} \left[\frac{3{R_i}^2}{R^2} - 1 \right]$$

The cross-sectional area A_R can be obtained from database tables. The partial derivatives respect to s/c state vector and the C_R coefficient are calculated straightforward dividing the perturbing acceleration by C_R.

5.10.2 GPS ROCK-models

For GPS satellites special solar radiation pressure models were developed by Rockwell international, based on the material properties of the spacecraft. The resulting models were called the GPS Rock-models. In NAPEOS the ROCK models are implemented and may be used. The amplitudes and phases of the different constituents of the model parameters are introduced through a file and thus different models may be used. This is convenient since different models were derived and published.

The general formulation of these models is

$$m\ddot{r}_i = \sum_{n=0}^{7} A_{in} B_{in} \sin\left(B_{in}\Phi + C_{in}\right)$$

where $m\ddot{r}_i$ are the components on this perturbation. The partial w.r.t. s/c position is:

$$m\frac{\partial \ddot{r}_i}{\partial r_j} = \left(\sum_{n=0}^{7} A_{in} B_{in} \cos\left(B_{in}\Phi + C_{in}\right) \right) \frac{\partial \Phi}{\partial r_j}$$

The formulae for the T10 version of the ROCK model, for the GPS-block I satellites, are:

$$X = -4.55\sin\Phi + 0.08\sin\left(2\Phi + 0.9\right) - 0.06\cos\left(4\Phi + 0.08\right) + 0.08$$

$$Z = -4.54\cos\Phi + 0.20\sin\left(2\Phi - 0.3\right) - 0.03\sin 4\Phi$$

where Φ is the angle between the Sun ant +Z axis of the satellite, measured in radians, and it is handled by the module OPst_STX

X and Z are given in 10-5 Newton's.

The formulae for the T20 version of the ROCK model, for the GPS-block II and IIA satellites, are:

$$X = -4.55 \sin\Phi + 0.16 \sin 3\Phi + 0.10 \sin 5\Phi - 0.07 \sin 7\Phi$$

$$Z = -8.43 \cos\Phi$$

The formulae for T30 version of the ROCK model, for the GPS-block IIR satellites, are:

$$X = -11.0 \sin\Phi - 0.2 \sin 3\Phi + 0.2 \sin 5\Phi$$

$$Z = -11.3 \cos\Phi + 0.1 \cos 3\Phi + 0.2 \cos 5\Phi$$

The evaluation of these forces is handled by the module OPrd_Radiation.f90

It should be noted, however, that in recent years it has become clear that these models are of limited value for precise POD applications. Equivalent, if not even better, results are obtained by ignoring the solar radiation models completely but just estimating parameters that absorb the solar radiation pressure forces very well, see RD-36

5.11 Aerodynamic forces

Though the atmospheric density at satellite altitude is low the drag remains as an important perturbation. This perturbing force can be written as follows (RD-8)

$$\Delta\vec{F} = -\frac{1}{2}\rho v^2 S_{ref}\left(c_d \vec{u}_d + c_n \vec{u}_n + c_l \vec{u}_l\right)$$

where

ρ is atmospheric density, as defined for the case of free molecular flow

v is velocity relative to the atmosphere

S_{ref} is reference surface of the satellite

$\left[c_d \; c_n \; c_l\right]$ are drag, lateral and lift coefficients respectively

$\vec{u}_d = -\frac{\vec{v}_w}{\left|\vec{v}_w\right|}$ where $\vec{v}_w$ is the s/c velocity , corrected with wind local velocity

$$\vec{u}_n = \frac{\vec{u}_d \times \vec{r}}{\left|\vec{u}_d \times \vec{r}\right|}$$

$$\vec{u}_l = \vec{u}_n \times \vec{u}_d$$

Usually effective surfaces, including the coefficients, for different satellite attitude are used.

5.12 Earth albedo and infra-red radiation pressure

The effect of the Earth albedo and the infra-red radiation pressure can be modelled as the acceleration due to a Earth area element ΔA:

$$\Delta\vec{a} = [C_{alb}\gamma I_s \cos\Psi_s + C_{ir} I_{ir}]\frac{\Delta A}{\pi d^2 c}\left(\frac{A_R}{M}\right)\cos\theta\,\vec{n}_1 \qquad \text{Eq. 5.5}$$

where

C_{alb} and C_{ir} are albedo and infra-red reflectivity coefficients

γ is the Earth albedo coefficient, i.e. fractions of incoming solar radiation re-emitted in visible region of spectrum

Ψ_s is angle between the element surface normal and the radio-vector to the Sun

I_s is re-radiation flux from sun at earth distance

I_{ir} is radiation flux re-emitted in infra-red region of spectrum

d is distance from spacecraft to the element

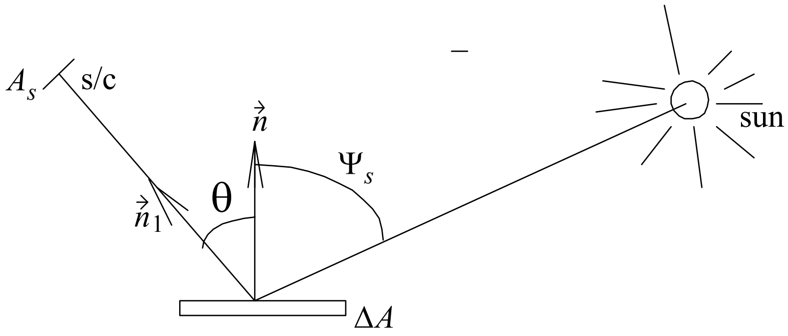
A_s is cross-section perpendicular to the element

m is the mass of the spacecraft

θ is the angle between the normal to ΔA and the direction to the spacecraft

c is the speed of light

$\vec{n}$ is the unit vector from the surface to the spacecraft



Both the albedo coefficient and the infra-red radiation flux are assumed to depend on latitude.

The evaluation of equation (5.5) is carried out with a numerical approach, whose details can be found in RD-11.

5.13 Long manoeuvres

The long manoeuvre for a satellite is given for its acceleration profile. This profile is used for the integration of Newton's equation. In some cases the manoeuvre is considered as impulsive, that is, a $\Delta\vec{v}$ is applied to the velocity vector. This increment of velocity is calculated as

$$\Delta\vec{v} = \int_{t_{begin}}^{t_{end}} \vec{a}(t)dt$$

where

$\vec{a}$ is the acceleration profile

t_{begin} and t_{end} are manoeuvre start and end time, respectively

The time when this $\Delta\vec{v}$ is applied is calculated as follows

$$t_{apply} = \frac{\int_{t_{begin}}^{t_{end}} \vec{a}(t)tdt}{\int_{t_{begin}}^{t_{end}} \vec{a}(t)dt}$$

5.14 Empirical accelerations

Empirical accelerations are used to compensate model errors. For low flying satellites typically the Earth gravity field and/or the atmospheric drag forces are the dominating error sources whereas for high-altitude satellites (e.g. GNSS) the solar radiation pressure force constitutes the main disturbance.

To compensate for the model omissions typically a three directional decomposition is used. Examples are radial-, along-, and cross-track for the LEO satellites and a Satellite-Sun-Earth based reference frame for the GNSS satellites. In NAPEOS the parameters to compensate for the gravitational and atmospheric forces are called “CPR’s” whereas the solar radiation pressure force parameters are called “CODE” parameters.

The 3-axis used for the NAPEOS CPR parameters are the radial direction $\vec{u}_r$, the transversal direction $\vec{u}_t$ defined as perpendicular to the geocentric position vector and the velocity vector. The third direction, called normal $\vec{u}_n$, is perpendicular to the other two. This system is actually orthogonal and for nearly circular orbits it is approximate to the previous described system.

The empirical accelerations are modelled as a combination of constant and periodic terms using the satellite argument of latitude as angular argument. Thus the period of the accelerations is one cycle per revolution hence the name CPR.

$$\begin{aligned}
\Delta\vec{a}_r &= \left(a_{r0} + a_{rc}\cos u + a_{rs}\sin u\right)\vec{u}_r \\
\Delta\vec{a}_t &= \left(a_{t0} + a_{tc}\cos u + a_{ts}\sin u\right)\vec{u}_t \\
\Delta\vec{a}_n &= \left(a_{n0} + a_{nc}\cos u + a_{ns}\sin u\right)\vec{u}_n
\end{aligned}$$

where u is the argument of latitude of the satellite.

The 3-axis used for the NAPEOS CODE parameters are the Satellite-Sun direction $\vec{u}_d$, the Y-axis, rotation axis of the solar panels, direction $\vec{u}_y$ defined as perpendicular to the Satellite-Sun direction and the Satellite-Earth direction. The third direction completing the right-handed system, called “B” (for “Bernese” direction) $\vec{u}_b$, is perpendicular to the other two.

These empirical accelerations are modelled as a combination of periodic of one cycle per revolution.

$$\begin{aligned}
\Delta\vec{a}_d &= \left(a_{d0} + a_{dc}\cos u + a_{ds}\sin u\right)\vec{u}_d \\
\Delta\vec{a}_y &= \left(a_{y0} + a_{yc}\cos u + a_{ys}\sin u\right)\vec{u}_y \\
\Delta\vec{a}_b &= \left(a_{b0} + a_{bc}\cos u + a_{bs}\sin u\right)\vec{u}_b
\end{aligned}$$

where in this case u is the argument of latitude of the satellite relative to the argument of latitude of the Sun in the orbital plane!

For both parameterisations, the nine coefficients that can be estimated can be given known a priori values and they can be estimated parameters in the orbit determination process.

6 PARAMETER ESTIMATION

6.1 The Estimation Problem

If at some time t_0 the satellite state vector X is known (X_0) and the forces acting on the satellite are known, then the satellite equations of motion can be integrated to determine the state vector of the satellite at any future time.

However, the initial state vector in never known exactly. Moreover, certain force models require physical parameters that are known only approximately, for example the satellite drag coefficient in aerodynamical forces, or the coefficients of the spherical harmonic expansion representation of the terrestrial gravity field. This is also the case with geophysical parameters that affect indirectly the equations of motion (like earth rotation and polar motion). Consequently, to determine the position of the satellite at a future time it is necessary that observations of the satellite are taken and used to obtain a better estimate of the satellite trajectory. The observational data, which will be subject to both systematic and random errors, will usually consist of measurements such as range, range-rate (doppler), azimuth, elevation or some other observable quantity. These measurements have to be corrected with models which are imperfect (e.g. tropospheric correction) and are usually taken from stations whose earth-fixed coordinates are not exactly known.

The problem of determining the best estimate of the satellite state vector (and optionally other parameters and geophysical and geodetic quantities) is referred to as *statistical orbit determination*. Sometimes the main interest is focused on the estimation of the geophysical and geodetic quantities themselves: this field is called *space geodesy*. Another variation of the problem is *simulation and covariance analysis* in which orbit determination strategies for a future mission are studied in terms of coverage, accuracy requirements and weighting of tracking data, orbit determination accuracy expectation, optimal arc-length...

The problem can be generalized as follows: given an initial state vector at time t_0 and the initial values of parameters to be estimated (not necessarily at t_0), together with their a-priori covariance matrix, and given a set of real or simulated observations (not necessarily after t_0) also with their initial covariance matrix, find the "best" estimate of the state vector at a future time and of the rest of the parameters, together with an a-posteriori parameter and observation covariance matrix. So, in matrix notation the a-priori values are:

$$X_0 = \begin{bmatrix} x_0 \\ y_0 \\ z_0 \\ x_0 \\ y_0 \\ z_0 \\ \alpha_1 \\ \vdots \\ \alpha_k \end{bmatrix}, \quad P_0 = \begin{bmatrix} \sigma_{11}{}^2 & \sigma_{12} & \cdots & \sigma_{1n} \\ \sigma_{21} & \sigma_{22}{}^2 & \cdots & \sigma_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ \sigma_{n1} & \sigma_{n2} & \cdots & \sigma_{nn}{}^2 \end{bmatrix} \qquad \text{Eq. 6.1}$$

$$Y = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_m \end{bmatrix}, \quad Q_0 = \begin{bmatrix} \sigma_1{}^2 & 0 & \cdots & 0 \\ 0 & \sigma_2{}^2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \sigma_m{}^2 \end{bmatrix} \qquad \text{Eq. 6.2}$$

X_0 being the matrix of estimated parameters or *extended initial state vector* (n is the total number of estimated parameters).

P_0 is the a-priori covariance matrix of the parameters in X_0. σ_{ii}^2 is the a-priori variance of parameter i, σ_{ij} is the a-priori covariance of parameters i and j, which is an indicator of their correlation or interdependence. For a mathematical definition of these statistical concepts see RD-12.

Y is the matrix containing the observations (m is the total number of observations, $m > n$) and Q_0 is the observation a-priori covariance matrix. We will assume here that the observations are a-priori not correlated, i.e. matrix Q_0 is diagonal; this is not necessarily true, but it is in general not easy to find an a-priori estimation of the observation covariance's. Let's call $\hat{X}(t)$ the best estimate of the extended state vector at time t, which is what we are trying to solve.

By far the most widely-used criterion to obtain in practice the "best" estimate or solution is to minimize the sum of the square of the *weighted residual observation errors*, that is, the square of the difference between the observation and the expected value computed from an observation model (*computed* observation), multiplied by a factor (*weight*) according to the observation importance and expected accuracy. These methods are called *least-square estimators,* see RD-9, RD-20, RD-21, RD-25, and RD-29.

In practice, observations are computed by evaluating the satellite state vector at the observation time and finding a geometric/kinematic relationship between the satellite position/velocity and the magnitude which is observed. We can then assume that the computed observations are a function of $\hat{X}(t)$, and define the residual observation vector as:

$$\varepsilon = \begin{bmatrix} y_1 - f_1\left(\hat{X}(t_1)\right) \\ y_2 - f_2\left(\hat{X}(t_2)\right) \\ \vdots \\ y_m - f_m\left(\hat{X}(t_m)\right) \end{bmatrix} \qquad \text{Eq. 6.3}$$

The quantity to be minimized (i.e. the sum of the square of the weighted residual observation errors) is called the *loss function* and can be expressed as:

$$J = \varepsilon^T \cdot Q_0^{-1} \cdot \varepsilon \qquad \text{Eq. 6.4}$$

Q_0^{-1} being the *weight matrix*. The weight matrix is necessary not only to give more importance to the most accurate observations but also to avoid adding magnitudes of different units (e.g. ranges and velocities). If the measurements are uncorrelated then Q_0 is diagonal:

$$Q_0^{-1} = \begin{bmatrix} \sigma_1^{-2} & 0 & \cdots & 0 \\ 0 & \sigma_2^{-2} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \sigma_m^{-2} \end{bmatrix} \qquad \text{Eq. 6.5}$$

The loss function described in Eq. 6.4 can be modified to account for the uncertainties of the a-priori values of the estimated parameters (see RD-26). This leads to the definition:

$$J = \varepsilon^T \cdot Q_0^{-1} \cdot \varepsilon + \Psi \qquad \text{Eq. 6.6}$$

where Ψ uses in some way (see below) information on the covariance of the $\hat{X}$ s. The problem is then to find the $\hat{X}$ which minimize *J*.

There are two major classes of least-squares estimators: batch and sequential (see RD-9, RD-20, RD-25, and RD-30.

A *batch* estimator updates the *extended state vector* X_0 (and optionally P_0 and Q_0) iteratively after a high enough number of observations (which define the *estimation arc*) has been collected after the epoch t_0. Once the process has converged to a best estimate of X_0 ($\hat{X}_0$), the satellite state vector can be propagated to any future time using as initial values the ones from $\hat{X}_0$.

In a *sequential* estimator observations are processed as soon as they are received, and the *extended state vector X* and its covariance matrix *P* are propagated/updated with every new observation or small set of observations. The main application of these estimators is the operational real-time orbit determination.

Batch and sequential methods will be described later in this note.

6.2 Batch Estimation (BAHN)

The objective is to find a best estimate of X0, (and optionally of P0 and Q0). We have seen that the computed observations can be expressed as a geometric/kinematic function of the satellite state vector at the time of the observation. As long as the satellite state vector can be propagated to any time from the initial state vector, the computed observations are a function of the time. The residual observation matrix can then be written as:

$$\varepsilon(\hat{X}_0) = \begin{bmatrix} y_1 - f_1(\hat{X}_0, t_1) \\ y_2 - f_2(\hat{X}_0, t_2) \\ \vdots \\ y_m - f_m(\hat{X}_0, t_m) \end{bmatrix} \qquad \text{Eq. 6.7}$$

Let's define in this case the loss function in Eq. 6.6 as:

$$J(\hat{X}_0) = \varepsilon^T \cdot Q_0^{-1} \cdot \varepsilon + \Delta X_0^{\ T} \cdot P_0^{-1} \cdot \Delta X_0 \qquad \text{Eq. 6.8}$$

Where:

$$\Delta X_0 = \hat{X}_0 - X_0$$

An $\hat{X}_0$ has to be found which minimizes the loss function $J(\hat{X}_0)$. This is achieved by differentiating Eq. 6.8 with respect to the estimated parameters $\hat{X}_0$ and setting the resulting expression to zero. The problem has to be linearized in order to be solved. Let's assume that the difference between $\hat{X}_0$ and X_0 (ΔX_0) is small (i.e. the initial values of the estimated parameters are a good enough approximation of the optimal ones). Then the computed observations can be expressed as their first order Taylor expansion around X_0:

$$f_i(\hat{X}_0, t_i) = f_i(X_0, t_i) + \sum_{j=1}^{n} \frac{\partial f_i}{\partial \beta_j} \cdot \left(\hat{\beta}_j - \beta_j\right) \qquad \text{Eq. 6.9}$$

where the betas are all the estimated parameters, i.e. the elements of the *X_0* vector. The residuals can be expressed then as:

$$\varepsilon(\hat{X}_0) = \begin{bmatrix} y_1 - f_1(X_0, t_1) \\ y_2 - f_2(X_0, t_2) \\ \vdots \\ y_m - f_m(X_0, t_m) \end{bmatrix} - F \cdot (\hat{X}_0 - X_0) \qquad \text{Eq. 6.10}$$

F is the *matrix of observation equation coefficients*, which contains the partial derivatives of the computed observations with respect to the estimated parameters:

$$F = \begin{bmatrix} \frac{\partial f_1}{\partial x_0} & \frac{\partial f_1}{\partial y_0} & \frac{\partial f_1}{\partial z_0} & \frac{\partial f_1}{\partial \dot{x}_0} & \frac{\partial f_1}{\partial \dot{y}_0} & \frac{\partial f_1}{\partial \dot{z}_0} & \frac{\partial f_1}{\partial \alpha_1} & \cdots & \frac{\partial f_1}{\partial \alpha_k} \\ \frac{\partial f_2}{\partial x_0} & \frac{\partial f_2}{\partial y_0} & \frac{\partial f_2}{\partial z_0} & \frac{\partial f_2}{\partial \dot{x}_0} & \frac{\partial f_2}{\partial \dot{y}_0} & \frac{\partial f_2}{\partial \dot{z}_0} & \frac{\partial f_2}{\partial \alpha_1} & \cdots & \frac{\partial f_2}{\partial \alpha_k} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \cdots & \vdots \\ \frac{\partial f_m}{\partial x_0} & \frac{\partial f_m}{\partial y_0} & \frac{\partial f_m}{\partial z_0} & \frac{\partial f_m}{\partial \dot{x}_0} & \frac{\partial f_m}{\partial \dot{y}_0} & \frac{\partial f_m}{\partial \dot{z}_0} & \frac{\partial f_m}{\partial \alpha_1} & \cdots & \frac{\partial f_m}{\partial \alpha_k} \end{bmatrix} \qquad \text{Eq. 6.11}$$

Substituting Eq. 6.10 in Eq. 6.8 and differentiating with respect to $\hat{X}_0$ leads to the following iterative algorithm (called ***normal equations***):

$$X_0^{\,k+1} = \hat{X}_0^{\,k} + (P_0^{-1} + F^T \cdot Q_0^{-1} \cdot F)^{-1} \cdot (\Delta Y_k + P_0^{-1} \cdot (X_0 - \hat{X}_0^{\,k})) \qquad \text{Eq. 6.12}$$

$\hat{X}_0^{\,k}$ is the estimation of on iteration *k* (equal to *X_0* on the first iteration).

ΔY_k is the residual observation matrix calculated propagating $\hat{X}_0^{\,k}$:

$$\Delta Y_k = \begin{bmatrix} y_1 - f_1(\hat{X}_0, t_1) \\ y_2 - f_2(\hat{X}_0, t_2) \\ \vdots \\ y_m - f_m(\hat{X}_0, t_m) \end{bmatrix} \qquad \text{Eq. 6.13}$$

The following matrix:

$$N = P_0^{-1} + F^T \cdot Q_0^{-1} \cdot F \qquad \text{Eq. 6.14}$$

is called the ***normal matrix***. It can be shown that the inverse of the normal matrix N is the best estimation of the covariance matrix of the estimated parameters, that is:

$$\hat{P}_0 = N^{-1} = (P_0^{-1} + F^T \cdot Q_0^{-1} \cdot F)^{-1} \qquad \text{Eq. 6.15}$$

This matrix is called in short ***covariance matrix***. It is usual (see RD-9 and RD-30) to modify the above expression in this way:

$$\hat{P}_0 = \frac{J}{m} \cdot N^{-1} = \frac{J}{m} \cdot (P_0^{-1} + F^T \cdot Q_0^{-1} \cdot F)^{-1} \qquad \text{Eq. 6.16}$$

J being the value of the ***loss function*** (see below the definition of *RMS*) and *m* the number of observations.

6.2.1 Observation equation coefficients and variational partials

The calculation method of the different partial derivatives in *F* depends highly on the nature of the variable with respect to which the derivative has to be calculated (i.e. the corresponding estimated parameter). Some of them are zero because the observation does not depend on the parameter. Other calculations are straightforward; for example, if the observation is a one-way range from a ground station and the parameter in question is the one of the station coordinates, then the computed observation would be:

$$R = \sqrt{(x - X_s)^2 + (y - Y_s)^2 + (z - Z_s)^2} \qquad \text{Eq. 6.17}$$

where *(x,y,z)* is the position of the satellite at the time of the observation and *(Xs,Ys,Zs)* are the station coordinates. The partial derivative of *R* with respect to the parameter *Xs* would be:

$$\frac{\partial R}{\partial X_s} = \frac{X_s - x}{R} \qquad \text{Eq. 6.18}$$

The way to compute partial derivatives with respect to other parameters can be found in RD-9.

A special set of parameters are the so-called ***dynamic parameters***, which are the satellite initial state vector $(x_0, y_0, z_0, \dot{x}_0, \dot{y}_0, \dot{z}_0)$ and the force model unknowns. The partial derivative of an observation w.r.t. a dynamic parameter can be expressed as:

$$\frac{\partial f}{\partial \beta} = \begin{bmatrix} \frac{\partial f}{\partial x} & \frac{\partial f}{\partial y} & \frac{\partial f}{\partial z} & \frac{\partial f}{\partial \dot{x}} & \frac{\partial f}{\partial \dot{y}} & \frac{\partial f}{\partial \dot{z}} \end{bmatrix} \begin{bmatrix} \frac{\partial x}{\partial \beta} \\ \frac{\partial y}{\partial \beta} \\ \frac{\partial z}{\partial \beta} \\ \frac{\partial \dot{x}}{\partial \beta} \\ \frac{\partial \dot{y}}{\partial \beta} \\ \frac{\partial \dot{z}}{\partial \beta} \end{bmatrix} \qquad \text{Eq. 6.19}$$

The coefficients of the file matrix in the right-hand side of Eq. 6.17 are the partial derivatives of the computed observation w.r.t. the satellite position and velocity at the time of observation, and can be easily evaluated for any kind of observation. For example, differentiating Eq. 6.18 w.r.t. *x* we obtain:

$$\frac{\partial R}{\partial x} = \frac{x - X_s}{R} \qquad \text{Eq. 6.20}$$

However, the coefficients of the column matrix in the right-hand side of Eq. 6.19 (i.e. the partial derivatives of the satellite position w.r.t. the dynamic parameter) have to be obtained by numerical integration of the *variational equations*. The column matrices for all the dynamic parameters to be estimated form the matrix of *position partials*:

$$X_m = \begin{bmatrix} \frac{\partial x}{\partial x_0} & \frac{\partial x}{\partial y_0} & \frac{\partial x}{\partial z_0} & \frac{\partial x}{\partial \dot{x}_0} & \frac{\partial x}{\partial \dot{y}_0} & \frac{\partial x}{\partial \dot{z}_0} & \frac{\partial x}{\partial \alpha_1} & \cdots & \frac{\partial x}{\partial \alpha_l} \\ \frac{\partial y}{\partial x_0} & \frac{\partial y}{\partial y_0} & \frac{\partial y}{\partial z_0} & \frac{\partial y}{\partial \dot{x}_0} & \frac{\partial y}{\partial \dot{y}_0} & \frac{\partial y}{\partial \dot{z}_0} & \frac{\partial y}{\partial \alpha_1} & \cdots & \frac{\partial y}{\partial \alpha_l} \\ \frac{\partial z}{\partial x_0} & \frac{\partial z}{\partial y_0} & \frac{\partial z}{\partial z_0} & \frac{\partial z}{\partial \dot{x}_0} & \frac{\partial z}{\partial \dot{y}_0} & \frac{\partial z}{\partial \dot{z}_0} & \frac{\partial z}{\partial \alpha_1} & \cdots & \frac{\partial z}{\partial \alpha_l} \end{bmatrix} \qquad \text{Eq. 6.21}$$

h=6+l being the number of dynamic parameters. Matrix X_m is evaluated by integrating the following equation of matrices called variational equations:

$$\ddot{X}_m = D1 \cdot X_m + D2 \cdot \dot{X}_m + A_f \qquad \text{Eq. 6.22}$$

being:

$$\dot{X}_m = \begin{bmatrix} \frac{\partial \dot{x}}{\partial x_0} & \frac{\partial \dot{x}}{\partial y_0} & \frac{\partial \dot{x}}{\partial z_0} & \frac{\partial \dot{x}}{\partial \dot{x}_0} & \frac{\partial \dot{x}}{\partial \dot{y}_0} & \frac{\partial \dot{x}}{\partial \dot{z}_0} & \frac{\partial \dot{x}}{\partial \alpha_1} & \cdots & \frac{\partial \dot{x}}{\partial \alpha_l} \\ \frac{\partial \dot{y}}{\partial x_0} & \frac{\partial \dot{y}}{\partial y_0} & \frac{\partial \dot{y}}{\partial z_0} & \frac{\partial \dot{y}}{\partial \dot{x}_0} & \frac{\partial \dot{y}}{\partial \dot{y}_0} & \frac{\partial \dot{y}}{\partial \dot{z}_0} & \frac{\partial \dot{y}}{\partial \alpha_1} & \cdots & \frac{\partial \dot{y}}{\partial \alpha_l} \\ \frac{\partial \dot{z}}{\partial x_0} & \frac{\partial \dot{z}}{\partial y_0} & \frac{\partial \dot{z}}{\partial z_0} & \frac{\partial \dot{z}}{\partial \dot{x}_0} & \frac{\partial \dot{z}}{\partial \dot{y}_0} & \frac{\partial \dot{z}}{\partial \dot{z}_0} & \frac{\partial \dot{z}}{\partial \alpha_1} & \cdots & \frac{\partial \dot{z}}{\partial \alpha_l} \end{bmatrix}$$

$$\ddot{X}_m = \begin{bmatrix} \frac{\partial \ddot{x}}{\partial x_0} & \frac{\partial \ddot{x}}{\partial y_0} & \frac{\partial \ddot{x}}{\partial z_0} & \frac{\partial \ddot{x}}{\partial \dot{x}_0} & \frac{\partial \ddot{x}}{\partial \dot{y}_0} & \frac{\partial \ddot{x}}{\partial \dot{z}_0} & \frac{\partial \ddot{x}}{\partial \alpha_1} & \cdots & \frac{\partial \ddot{x}}{\partial \alpha_l} \\ \frac{\partial \ddot{y}}{\partial x_0} & \frac{\partial \ddot{y}}{\partial y_0} & \frac{\partial \ddot{y}}{\partial z_0} & \frac{\partial \ddot{y}}{\partial \dot{x}_0} & \frac{\partial \ddot{y}}{\partial \dot{y}_0} & \frac{\partial \ddot{y}}{\partial \dot{z}_0} & \frac{\partial \ddot{y}}{\partial \alpha_1} & \cdots & \frac{\partial \ddot{y}}{\partial \alpha_l} \\ \frac{\partial \ddot{z}}{\partial x_0} & \frac{\partial \ddot{z}}{\partial y_0} & \frac{\partial \ddot{z}}{\partial z_0} & \frac{\partial \ddot{z}}{\partial \dot{x}_0} & \frac{\partial \ddot{z}}{\partial \dot{y}_0} & \frac{\partial \ddot{z}}{\partial \dot{z}_0} & \frac{\partial \ddot{z}}{\partial \alpha_1} & \cdots & \frac{\partial \ddot{z}}{\partial \alpha_l} \end{bmatrix}$$

$\ddot{X}_m$ is the matrix of *acceleration partials* and $\dot{X}_m$ the matrix of *velocity partials*.

The coefficients of A_f are the direct derivatives of the acting accelerations w.r.t. the force parameters.

$$A_f = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & \frac{\partial a_x}{\partial \alpha_1} & \cdots & \frac{\partial a_x}{\partial \alpha_l} \\ 0 & 0 & 0 & 0 & 0 & 0 & \frac{\partial a_y}{\partial \alpha_1} & \cdots & \frac{\partial a_y}{\partial \alpha_l} \\ 0 & 0 & 0 & 0 & 0 & 0 & \frac{\partial a_z}{\partial \alpha_1} & \cdots & \frac{\partial a_z}{\partial \alpha_l} \end{bmatrix}$$

D1 and *D2* are given by:

$$D1 = \begin{bmatrix} \frac{\partial a_x}{\partial x} & \frac{\partial a_x}{\partial y} & \frac{\partial a_x}{\partial z} \\ \frac{\partial a_y}{\partial x} & \frac{\partial a_y}{\partial y} & \frac{\partial a_y}{\partial z} \\ \frac{\partial a_z}{\partial x} & \frac{\partial a_z}{\partial y} & \frac{\partial a_z}{\partial z} \end{bmatrix}, \quad D2 = \begin{bmatrix} \frac{\partial a_x}{\partial \dot{x}} & \frac{\partial a_x}{\partial \dot{y}} & \frac{\partial a_x}{\partial \dot{z}} \\ \frac{\partial a_y}{\partial \dot{x}} & \frac{\partial a_y}{\partial \dot{y}} & \frac{\partial a_y}{\partial \dot{z}} \\ \frac{\partial a_z}{\partial \dot{x}} & \frac{\partial a_z}{\partial \dot{y}} & \frac{\partial a_z}{\partial \dot{z}} \end{bmatrix}$$

D1 and *D2* are the derivatives of the acting accelerations on the spacecraft w.r.t. the satellite position and velocity, respectively. Eq. 6.22 is a system of second order differential equations, with the initial conditions:

$$X_m^0 = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & \cdots & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & \cdots & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & \cdots & 0 \end{bmatrix}, \quad \dot{X}_m^0 = \begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 0 & 0 & \cdots & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & \cdots & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & \cdots & 0 \end{bmatrix}$$

These variational equations can be integrated numerically with the methods described in section 1. The contribution of each of the accelerations acting on the satellite must be evaluated at every integration step in order to compute matrices *D1*, *D2* and A_f. This evaluation is rather cumbersome (see RD-9 and RD-27) and is out of the scope of this note. In some cases the contribution of some of the forces can be ignored or calculated approximately. In the absence of atmospheric drag matrix *D2* is zero because no other force depends on the satellite velocity. The solution of the variational equations is matrix X_m (and $\dot{X}_m$ also), which is what we need in order to calculate the derivatives with respect to dynamic parameters in *F*. The elements of X_m and $\dot{X}_m$ are also called *variational partials*. A subset of elements from X_m and $\dot{X}_m$ form the so-called *state transition matrix*:

$$\phi(t) = \begin{bmatrix} \frac{\partial x}{\partial x_0} & \frac{\partial x}{\partial y_0} & \frac{\partial x}{\partial z_0} & \frac{\partial x}{\partial \dot{x}_0} & \frac{\partial x}{\partial \dot{y}_0} & \frac{\partial x}{\partial \dot{z}_0} \\ \frac{\partial y}{\partial x_0} & \frac{\partial y}{\partial y_0} & \frac{\partial y}{\partial z_0} & \frac{\partial y}{\partial \dot{x}_0} & \frac{\partial y}{\partial \dot{y}_0} & \frac{\partial y}{\partial \dot{z}_0} \\ \frac{\partial z}{\partial x_0} & \frac{\partial z}{\partial y_0} & \frac{\partial z}{\partial z_0} & \frac{\partial z}{\partial \dot{x}_0} & \frac{\partial z}{\partial \dot{y}_0} & \frac{\partial z}{\partial \dot{z}_0} \\ \frac{\partial \dot{x}}{\partial x_0} & \frac{\partial \dot{x}}{\partial y_0} & \frac{\partial \dot{x}}{\partial z_0} & \frac{\partial \dot{x}}{\partial \dot{x}_0} & \frac{\partial \dot{x}}{\partial \dot{y}_0} & \frac{\partial \dot{x}}{\partial \dot{z}_0} \\ \frac{\partial \dot{y}}{\partial x_0} & \frac{\partial \dot{y}}{\partial y_0} & \frac{\partial \dot{y}}{\partial z_0} & \frac{\partial \dot{y}}{\partial \dot{x}_0} & \frac{\partial \dot{y}}{\partial \dot{y}_0} & \frac{\partial \dot{y}}{\partial \dot{z}_0} \\ \frac{\partial \dot{z}}{\partial x_0} & \frac{\partial \dot{z}}{\partial y_0} & \frac{\partial \dot{z}}{\partial z_0} & \frac{\partial \dot{z}}{\partial \dot{x}_0} & \frac{\partial \dot{z}}{\partial \dot{y}_0} & \frac{\partial \dot{z}}{\partial \dot{z}_0} \end{bmatrix} \qquad \text{Eq. 6.23}$$

which is formed by the partial derivatives of the satellite state vector w.r.t. the initial state vector at the epoch t_0. This matrix will always be available because at least the six elements of the satellite initial state vector must always be estimated. An application of this matrix will be seen later.

6.2.2 Iterative Sequence and Post-Convergence Analysis

The scheme of the batch estimator, based on the algorithm given by Eq. 6.12, becomes clear now: a propagator propagates the satellite position and velocity from X_0, at consecutive steps of the time arc defined by the available observational data. These positions and velocities are stored. The propagator integrates also the variational equations at the same time steps and the resulting variational partials X_m and $\dot{X}_m$ are also stored.

The observations are then sequentially processed: for each observation, a computed observation is generated as a function f of the satellite position and/or velocity at the observation time. Normally this time will not coincide with one of the integration steps, so an interpolation algorithm will have to be applied. The difference between the observation and the computed observation (the *residual* y_i-f_i) is then calculated. Also for each observation, the observation equation coefficients (rows of matrix F) are calculated as described above. This will also require interpolating the variational partials.

Once all observations have been processed matrices ΔY and F are formed. The normal matrix N (Eq. 6.14) is then calculated and inverted. All these matrices are then fed into the normal equations (Eq. 6.12) and a new guess of the initial state vector is obtained, $\hat{X}_0^1$.

Note that to solve normal equations, it is not necessary to invert normal matrix. Normal equations can be solved without performing such inversion in a much faster process. However, if the normal matrix is not inverted, the covariance matrix in Eq. 6.16 cannot be obtained. In any case, covariance matrix is only interesting to be obtained in last iteration of the estimation process, in the intermediate iterations, its knowledge does not provide any major information.

Once the normal equations have been solved, the propagator propagates then the new estimation of the initial state vector and the whole process is repeated until convergence is reached. The process terminates when the difference between two consecutive estimations tends to zero, i.e. the loss function J (Eq. 6.8) reaches a minimum. Although the loss function does not appear in the normal equations, it must be evaluated at every iteration because it is the quantity whose variation determines the convergence or divergence of the process (for example if the change of the loss function in percentage is less than a given threshold iterations are stopped). Another useful quantity to check the convergence of the process is the root-mean-square of weighted residuals or RMS:

$$RMS = \sqrt{J/m} \qquad \text{Eq. 6.24}$$

That is, the square root of the loss function (its value on the last iteration) divided by the number of observations. The RMS is an indication of whether the a-priori covariances have been guessed correctly and, in theory, it should converge to a value of unity. If the process converges properly, none of the observation weighted residuals computed on each iteration should be much greater than the RMS. Normally, some observations are affected by systematic errors: it is common practice then to eliminate after each iteration those observations whose weighted residual is greater than a multiple (e.g. 3) of the RMS. A very high number of rejected observations could indicate an incorrect guess of the initial values of the estimated parameters or incorrect a-priori observation weights.

After convergence the best estimate of the covariance matrix of the estimated parameters, $\hat{P}_0$, is calculated using Eq. 6.15or Eq. 6.16.

The covariance matrix of the satellite state vector (position and velocity) at t_0, which is a (6x6) submatrix of $\hat{P}_0$, can be propagated to a future time by means of the *state transition matrix* (Eq. 6.23):

$$\hat{P}(t) = \phi(t) \cdot \hat{P}_0 \cdot \phi(t)^T \qquad \text{Eq. 6.25}$$

where $\hat{P}(t)$ denotes the best estimate of the (6x6) satellite state vector covariance matrix in a future time *t*.

Another interesting magnitude is the correlation coefficient of the estimated parameters, defined as:

$$Cij = \frac{\hat{P}_{0ij}}{\sqrt{\hat{P}_{0ii} \cdot \hat{P}_{0jj}}} \qquad \text{Eq. 6.26}$$

where *Cij* is the correlation coefficient between parameters *i* and *j*. Correlation coefficients range from -1 to +1; either extreme value indicates that the two parameters are completely dependent and one may be eliminated from the estimation process.

The covariance matrix, apart from giving the variances and covariances of the unknowns, can also be used to compute the a-posteriori (after adjustment) variances of the observations (RD-9). After convergence the covariance matrix of the observations, which is also the covariance matrix of the residuals, is:

$$\hat{Q}_0 = F \cdot P_0 \cdot F^T \qquad \text{Eq. 6.27}$$

The final orbit is propagated using the $\hat{X}_0$ obtained in the last iteration. The final residuals can be computed directly using the final orbit or evaluated using Eq. 6.10:

$$\Delta Y^{k+1} = \Delta Y^k - F \cdot (\hat{X}_0{}^{k+1} - \hat{X}_0{}^k) \qquad \text{Eq. 6.28}$$

where $\hat{X}_0^{k+1}$ is the final estimate of the state vector and F and ΔY^k are the values on the last iteration of the matrix of observation coefficients and of the residuals, respectively. Normally matrix F is not stored after each iteration and the residuals after the last iteration are calculated directly from the final orbit. We'll see below another application of Eq. 6.28.

6.2.3 Normal Equations Solutions

In order to solve the normal equations the normal matrix N may be (or not) inverted. Here we can make use of the fact that a normal matrix is always symmetric and positive-definite. These two properties make it possible that the normal matrix can be decomposed and inverted using the *Choleski's decomposition method* which we will briefly explain below. The theory given here is based on RD-9 and RD-25.

Let's express Eq. 6.12 as:

$$N \cdot X = B \qquad \text{Eq. 6.29}$$

being:

$$X = \hat{X}_0^{k+1} - \hat{X}_0^{k}$$

$$N = P_0^{-1} + F^T \cdot Q_0^{-1} \cdot F$$

$$B = P_0^{-1} \cdot (X_0 - \hat{X}_0^{k}) + F^T \cdot Q_0^{-1} \cdot \Delta Y$$

N is a *(nxn)* matrix, X is a *(nx1)* matrix and B is a *(nx1)* matrix. The first step of Choleski's decomposition consists of factorizing N as:

$$N = L \cdot L^T \qquad \text{Eq. 6.30}$$

where L and L^T are lower and upper triangular matrices, respectively. L is also called a square root of N. Every positive-definite matrix has a square root. The elements of L can be obtained by expanding Eq. 6.30 and solving term by term (see RD-9 and RD-25). The normal equations can be written then as:

$$L \cdot (L^T \cdot X) = B \qquad \text{Eq. 6.31}$$

Defining matrix R as:

$$R = L^T \cdot X \qquad \text{Eq. 6.32}$$

Eq. 6.29 can be expressed as:

$$L \cdot R = B \qquad \text{Eq. 6.33}$$

The elements of *R* can be obtained using a forward recursion relation in Eq. 6.33. After *R* has been solved, the elements of *X* are obtained using a backward recursion in Eq. 6.32. Choleski's decomposition can be very efficiently implemented in a computer program to both solving the normal equations and finding the inverse of the normal matrix.

$$N = L \cdot L^T \quad \rightarrow \quad N^{-1} = L^{-1} \cdot (L^T)^{-1} = L^{-1} \cdot (L^{-1})^T \qquad \text{Eq. 6.34}$$

Thus, when the lower decomposed matrix has been inverted, the inverse of the original matrix can be easily obtained using Eq. 6.34.

6.2.4 'Consider' Parameters and Covariance Analysis

In some applications it is interesting to evaluate how uncertainties in some parameters affect the estimation process, without estimating those parameters. These parameters are called *consider parameters*: they do not affect the final value of the estimated parameters, but they do affect the estimate of their covariance matrix.

Let's assume first that all parameters will be estimated and calculate the corresponding normal matrix *N*. The normal equations can be expressed as in Eq. 6.29. If we split now the increment in the extended state vector, *X*, in estimated (*e*) and consider (*c*) parameters and we assume that a-priori *e* and *c* parameters are uncorrelated, Eq. 6.29 can be expressed in the form:

$$\begin{bmatrix} N_e & N_{ec} \\ N_{ec}{}^T & N_c \end{bmatrix} \cdot \begin{bmatrix} X_e \\ X_c \end{bmatrix} = \begin{bmatrix} B_e \\ B_c \end{bmatrix} \qquad \text{Eq. 6.35}$$

where:

$$N_e = P_{0e}{}^{-1} + F_e{}^T \cdot Q_0{}^{-1} \cdot F_e$$

$$N_{ec} = F_e{}^T \cdot Q_0{}^{-1} \cdot F_c$$

$$N_c = P_{0c}{}^{-1} + F_c{}^T \cdot Q_0{}^{-1} \cdot F_c$$

Expanding Eq. 6.35 the estimated parameters can be expressed as:

$$X_e = N_e{}^{-1} \cdot (B_e - N_{ec} \cdot X_c) \qquad \text{Eq. 6.36}$$

Let's now define the *sensitivity matrix* as:

$$S = \frac{\partial X_e}{\partial X_c} = -N_e^{-1} \cdot N_{ec} \qquad \text{Eq. 6.37}$$

If we now take into account that consider parameters do not affect the final value of the estimated parameters, it can be shown (RD-25) that Eq. 6.36 becomes:

$$X_e = N_e^{-1} \cdot B_e \qquad \text{Eq. 6.38}$$

and the covariance matrix of the estimated parameters, affected by the consider parameters is (RD-9):

$$\hat{P}_{0e}^{\ *} = \hat{P}_{0e} + S \cdot P_{0c} \cdot S^T = N_e^{-1} + (N_e^{-1} \cdot N_{ec}) \cdot P_{0c} \cdot (N_{ec}^{\ T} \cdot N_e^{-1}) \qquad \text{Eq. 6.39}$$

where P_{0c} is the a-priori covariance matrix of the consider parameters. Note that Eq. 6.38 is exactly the same as the normal equations (Eq. 6.12): consider parameters don't play any role here.

The practical application of Eq. 6.38 and Eq. 6.39 is the following: the matrix of observation equation coefficients, F, is calculated for both estimated and considered parameters and submatrices N_e and N_{ec} of the normal matrix N are calculated using F and the a-priori covariance matrix of all considered and estimated parameters (Eq. 6.14). Matrix N_e is then inverted and used to calculate the increment in the state vector of estimated parameters X in the usual way (using Eq. 6.12or Eq. 6.38). After convergence, the covariance matrix of the estimated parameters, $\hat{P}_{0e}$, is calculated using Eq. 6.15. This covariance matrix is then corrected to account for the consider parameters using Eq. 6.39.

The above described process is the basis of covariance analysis, which is the study of how coverage and accuracy of observational data, estimation arc length and uncertainties in model parameters; affect the accuracy of the orbit determination process. For this purpose, a satellite tracking strategy is selected (for example, in an earth-observation mission, a polar station generating range and range-rate microwave observations whenever the satellite is visible), observations are simulated according to this strategy together with an a-priori observation covariance matrix Q_0. An a-priori uncertainty (covariance matrix) of the parameters that will not normally be estimated during a normal orbit determination arc (for example, tidal or gravity coefficients, station coordinates...) is evaluated, P_{0c}. These parameters are taken as consider parameters, an arc length is selected, and the process described above is followed in order to obtain the covariance matrix of the parameters that will be estimated during a normal orbit determination arc, $\hat{P}_{0e}{}^{*}$. An analysis of this matrix gives an estimate of what accuracy can be expected in the orbit determination scenario under study.

6.2.5 Epoch dependent parameters estimation

Especially when dealing with GNSS data, the estimation of epoch dependent parameters is widely used in the orbit determination process.

The parameters considered as epoch dependent parameters are those that only affect the estimation of the parameters present at the same estimation epoch. For instance, GNSS clocks are normally estimated in a snapshot process; this means that each estimated clock bias in the clock history of a certain clock in the process does not depend of the rest of the clock history.

As the parameters are estimated per epoch, the parameters can only affect the estimation of the parameters that happen at the same epoch. Thus, all the parameters at one single epoch can be eliminated from the normal equations.

By eliminating the epoch dependent parameters epoch by epoch from the normal equations, the final linear system to solve has a much smaller size.

In a similar manner as for Eq. 6.35, If we split now the estimated parameters, in non-epoch-dependent (*e*) and epoch dependent parameters (*epoch*) parameters and we assume that a-priori epochs are uncorrelated among themselves, Eq. 6.29 can be expressed in the form:

$$\begin{bmatrix} N_e & (N_e^{epoch1})^T & \cdots & (N_e^{epochi})^T & \cdots & (N_e^{epochp})^T \\ N_e^{epoch1} & N^{epoch1} & \cdots & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ N_e^{epochi} & 0 & \cdots & N^{epochi} & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ N_e^{epochp} & 0 & \cdots & 0 & \cdots & N^{epochp} \end{bmatrix} \cdot \begin{bmatrix} X_e \\ X^{epoch1} \\ \vdots \\ X^{epochi} \\ \vdots \\ X^{epochp} \end{bmatrix} = \begin{bmatrix} B_e \\ B^{epoch1} \\ \vdots \\ B^{epochi} \\ \vdots \\ B^{epochp} \end{bmatrix} \qquad \text{Eq. 6.40}$$

being p the number of epochs to compute.

This is the system to solve:

$$\begin{aligned} & N_e \cdot X_e + \sum_{i=1}^{p} (N_e^{epochi})^T \cdot X^{epochi} = B_e \\ & N_e^{epochi} \cdot X_e + N^{epochi} \cdot X^{epochi} = B^{epochi}, \quad \forall i \in [1, p] \end{aligned} \qquad \text{Eq. 6.41}$$

Eliminating the epoch by epoch, and introducing their contribution

$$\begin{aligned} &N_e^* \cdot X_e = B_e^* \\ &N^{epochi} \cdot X^{epochi} = B^{epochi} - N_e^{epochi} \cdot X_e, \quad \forall i \in [1, p] \end{aligned} \qquad \text{Eq. 6.42}$$

being:

$$N_e^* = N_e - \sum_{i=1}^{p} \left((N_e^{epochi})^T \cdot (N^{epochi})^{-1} \cdot N_e^{epochi} \right)$$

$$B_e^* = Be - \sum_{i=1}^{p} \left((N^{epochi})^{-1} \cdot B^{epochi} \right)$$

Thus, the epoch dependent parameter are eliminated and set in the non-epoch dependent part of the normal equations. Once all epochs have been eliminated, the resulting normal equations in Eq. 6.41 (first part of them) are solved to obtain the estimate of the non-epoch dependent parameters. With such estimate X_e, the second part of those normal equations is solved, obtaining the solution for the epoch dependent parameters.

Therefore, as the parameters can be eliminated epoch by epoch, one single epoch must be kept in memory while solving normal equations. However, to solve epoch dependent parameters afterwards, the matrices and must be re-accumulated epoch by epoch in order to solve the epoch dependent parameters for each epoch.

6.3 Normal Equation Stacking and Multi-Arc Methods

Some parameters can only be estimated ("observed") processing observations over a long span of time (from months to years), for example station coordinates or geopotential coefficients. It is not practical in these cases to process a unique long orbit determination arc because this would require long computation times, inverting large matrices and having to deal with huge observation datasets. Moreover, it can also be interesting to estimate this kind of parameters based on using observations of several satellites.

6.3.1 Method 1 (Old multiarc, no longer used in NAPEOS)

This method is based on [Eq. 6.25] and [Eq. 6.24].

Let's assume then that we have a number *k* of arcs containing observations, not necessarily of the same satellite. We are looking for a solution that gives the best estimate of *k* sets of parameters associated only with individual arcs (satellite state vector, drag coefficient, observation bias...) and a set of geodetic parameters that are common to all arcs (station coordinates, geopotential coefficients).

The normal equations are also applicable to this case, but their direct solution is not optimal because it involves a large number of parameters and a large number of observations. A more convenient solution to the problem is then necessary. Eq. 6.29 can be expressed as:

$$N \cdot \begin{bmatrix} X_1 \\ X_2 \\ \vdots \\ X_k \\ X_G \end{bmatrix} = \begin{bmatrix} B_1 \\ B_2 \\ \vdots \\ B_k \\ B_G \end{bmatrix} \qquad \text{Eq. 6.43}$$

where X_i is the matrix of estimated parameters associated with arc i and X_G is the matrix of geodetic parameters common to all arcs.

Let's assume now that the a-priori information of arc and common parameters is independent, which means that the a-priori covariance matrix of the estimated parameters is:

$$P_0 = \begin{bmatrix} P_{01} & 0 & \cdots & 0 & 0 \\ 0 & P_{02} & \cdots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \cdots & P_{0k} & 0 \\ 0 & 0 & \cdots & 0 & P_{0G} \end{bmatrix}, \quad P_0^{-1} = \begin{bmatrix} P_{01}^{-1} & 0 & \cdots & 0 & 0 \\ 0 & P_{02}^{-1} & \cdots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \cdots & P_{0k}^{-1} & 0 \\ 0 & 0 & \cdots & 0 & P_{0G}^{-1} \end{bmatrix}$$

It is also valid to assume that the observations of different arcs are uncorrelated, that is:

$$Q_0 = \begin{bmatrix} Q_{01} & 0 & \cdots & 0 \\ 0 & Q_{02} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & Q_{0k} \end{bmatrix}, \quad Q_0^{-1} = \begin{bmatrix} Q_{01}^{-1} & 0 & \cdots & 0 \\ 0 & Q_{02}^{-1} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & Q_{0k}^{-1} \end{bmatrix}$$

The matrix F of observation equation coefficients has the form:

$$F = \begin{bmatrix} F_1 & 0 & \cdots & 0 & F_{1G} \\ 0 & F_2 & \cdots & 0 & F_{2G} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \cdots & F_k & F_{kG} \end{bmatrix}$$

where F_i are the partial derivatives of the observations of arc i with respect to the parameters of arc i, and F_{iG} are the partial derivatives of the observations of arc i with respect to the common parameters. The rest of the elements of F are zero because the observations of one arc do not depend of the parameters of any other arc. The normal matrix has the following *arrow* form:

$$N = \begin{bmatrix} N_1 & 0 & \cdots & 0 & N_{1G} \\ 0 & N_2 & \cdots & 0 & N_{2G} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \cdots & N_K & N_{kG} \\ N_{1G}{}^T & N_{2G}{}^T & \cdots & N_{kG}{}^T & N_{GG} \end{bmatrix}$$

where:

$$\begin{aligned} N_i &= P_{0i}{}^{-1} + F_i^T \cdot Q_{0i}{}^{-1} \cdot F_i \\ N_{iG} &= F_i^T \cdot Q_{0i}{}^{-1} \cdot F_{iG} \\ N_{iG}{}^T &= F_{iG}{}^T \cdot Q_{0i}{}^{-1} \cdot F_i \\ N_{GG} &= P_{0G}{}^{-1} + \sum_{i=1}^{k} F_{iG}{}^T \cdot Q_{0i}{}^{-1} \cdot F_{iG} \end{aligned} \qquad \text{Eq. 6.44}$$

Eq. 6.43 can be then expressed as:

$$\begin{bmatrix} N_1 & 0 & \cdots & 0 & N_{1G} \\ 0 & N_2 & \cdots & 0 & N_{2G} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \cdots & N_K & N_{kG} \\ N_{1G}{}^T & N_{2G}{}^T & \cdots & N_{kG}{}^T & N_{GG} \end{bmatrix} \cdot \begin{bmatrix} X_1 \\ X_2 \\ \vdots \\ X_k \\ X_G \end{bmatrix} = \begin{bmatrix} B_1 \\ B_2 \\ \vdots \\ B_k \\ B_G \end{bmatrix} \qquad \text{Eq. 6.45}$$

where:

$$\begin{aligned} B_i &= F_i^T \cdot Q_{0i}{}^{-1} \cdot \Delta Y_i + P_{0i}{}^{-1} \cdot (X_{0i} - \hat{X}_{0i}{}^k) \\ B_G &= \sum_{i=1}^{k} F_{iG}{}^T \cdot Q_{0i}{}^{-1} \cdot \Delta Y_i + P_{0G}{}^{-1} \cdot (X_{0G} - \hat{X}_{0G}{}^k) \end{aligned} \qquad \text{Eq. 6.46}$$

Expanding Eq. 6.45 and regrouping terms leads to:

$$X_i = N_i^{-1} \cdot (B_i - N_{iG} \cdot X_G)$$

$$X_G = \left(N_{GG} - \sum_{i=1}^{k} N_{iG}{}^T \cdot N_i^{-1} \cdot N_{iG} \right)^{-1} \cdot \left(B_G - \sum_{i=1}^{k} N_{iG}{}^T \cdot N_i^{-1} \cdot B_i \right) \qquad \text{Eq. 6.47}$$

being:

$$X_i = \hat{X}_{0i}{}^{k+1} - \hat{X}_{0i}{}^{k}$$

$$X_G = \hat{X}_{0G}{}^{k+1} - \hat{X}_{0G}{}^{k}$$

The solution to the problem can be obtained as follows:

1. Solve each arc separately estimating only the arc parameters but calculating the observation equation coefficients (matrices F_i and F_{iG}) for all parameters (arc and common). After convergence, the estimated state vector $\hat{X}_{0i}$, its a-priori value X_{0i} its initial covariance matrix, P_{0i}, and the last iteration value of the arc residuals ΔY_i, their a-priori covariance matrix Q_{0i} and F_i and F_{iG} are stored. Note that the a-priori values of the common parameters X_{0G} must be known on each arc in order to calculate F_{iG}. Moreover they must be the same for all arcs and the same as the a-priori value used in step 3.
2. After processing all of the arcs, matrices N_i, N_{iG} and N_{GG} are computed using Eq. 6.44. Then, matrices N_i are inverted and matrices B_i are calculated.
3. Eq. 6.47 is solved in one iteration using X_{0G} as a-priori value of the common parameters:

$$\hat{X}_{0G} = X_{0G} + \left(N_{GG} - \sum_{i=1}^{k} N_{iG}{}^T \cdot N_i^{-1} \cdot N_{iG} \right)^{-1} \cdot \left(\sum_{i=1}^{k} F_{iG}{}^T \cdot Q_{0i}^{-1} \cdot \Delta Y_i - \sum_{i=1}^{k} N_{iG}{}^T \cdot N_i^{-1} \cdot B_i \right)$$

   The a-posteriori covariance matrix of the common parameters is:

$$\hat{P}_{0G} = \left(N_{GG} - \sum_{i=1}^{k} N_{iG}{}^T \cdot N_i^{-1} \cdot N_{iG} \right)^{-1}$$

4. The estimated state vectors of each arc obtained in step 1 are corrected using Eq. 6.47:

$$\hat{X}_{0i}{}^{*} = \hat{X}_{0i} + N_i^{-1} \cdot (B_i - N_{iG} \cdot (\hat{X}_{0G} - X_{0G}))$$

The new covariance matrix of the arc parameters is:

$$\hat{P}_{0i}^{\ *} = N_i^{-1^*} = N_i^{-1} + (N_i^{-1} \cdot N_{iG}) \cdot N_{GG}^{\ -1} \cdot (N_{iG}^{\ T} \cdot N_i^{-1})$$

5. The residuals can be recomputed to take into account the new values of the estimated parameters using Eq. 6.28:

$$\Delta Y_i^* = \Delta Y_i - F_i \cdot (\hat{X}_{0i}^{\ *} - \hat{X}_{0i}) - F_{iG} \cdot (\hat{X}_{0G} - X_{0G})$$

Step 1 implies the storage of one observation equation coefficient per observation and per parameter (arc and common). In addition a residual per observation must be also stored. It is clear then that for problems estimating a large number of parameters and/or processing a large number of observations the storage requirements would become enormous.

An alternative to step 1 is to solve each arc storing matrices N_i, N_{iG} and the contribution to N_{GG} and also B_i and the contribution to B_G, instead of ΔY_i, Q_{0i}, F_i and F_{iG}. The rest of the steps would be the same. Matrices N_i, N_i^{-1} and B_i have to be calculated anyway for each arc because they are necessary to solve the estimated parameters in Eq. 6.12. The storage requirements of this alternative are clearly orders of magnitude smaller. A disadvantage is that step 5 cannot be done anymore, that is, it is not possible to estimate what is the change in the observation residuals due to the improvement of the estimated parameters.

6.3.2 Method 2 (Old batusi, no longer used in NAPEOS)

Both arc and common parameters are estimated on each arc using no a-priori covariances for any of the parameters. The normal equations for each arc i can be expressed as:

$$\begin{bmatrix} N_a^i & N_{ag}^i \\ N_{ag}^{iT} & N_g^i \end{bmatrix} \cdot \begin{bmatrix} X_a^i \\ X_g^i \end{bmatrix} = \begin{bmatrix} B_a^i \\ B_g^i \end{bmatrix} \qquad \text{Eq. 6.48}$$

where:

$$N_a^i = F_a^{iT} \cdot Q_{0i}^{-1} \cdot F_a^i$$

$$N_{ag}^i = F_a^{iT} \cdot Q_{0i}^{-1} \cdot F_g^i$$

$$N_g^i = F_g^{iT} \cdot Q_{0i}^{-1} \cdot F_g^i$$

$$B_a^i = F_a^{iT} \cdot Q_{0i}^{-1} \cdot \Delta Y_i$$

$$B_g^i = F_g^{iT} \cdot Q_{0i}^{-1} \cdot \Delta Y_i$$

Expanding Eq. 6.48 and substituting we obtain:

$$N_g^{*i} \cdot X_g^i = B_g^{*i} \qquad \text{Eq. 6.49}$$

where:

$$N_g^{*i} = N_g^i - N_{ag}^{iT} \cdot (N_{ag}^i)^{-1} \cdot N_{ag}^i$$

$$B_g^{*i} = B_g^i - N_{ag}^{iT} \cdot (N_a^i)^{-1} \cdot B_a^i$$

Let's assume that after convergence has been reached for each arc, we do an extra iteration such that the improved common parameters are the same for all arcs. Eq. 6.49 would then become:

$$N_g^{*i} \cdot (\hat{X}_{0G} - \hat{X}_{0G}^i) = B_g^{*i} \qquad \text{Eq. 6.50}$$

Where $\hat{X}_{0G}$ are the improved parameters we are looking for and $\hat{X}_{0G}^i$ are the values obtained on each arc after convergence. Eq. 6.50 cannot be combined because the $\hat{X}_{0G}^i$ are different for each arc. Let's assume that the extra iteration is done from an a-priori value X_{0G}. Eq. 6.50 can be rewritten as:

$$N_g^{*i} \cdot (\hat{X}_{0G} - X_{0G} + X_{0G} - \hat{X}_{0G}^i) = B_g^{*i}$$

or:

$$N_g^{*i} \cdot (\hat{X}_{0G} - X_{0G}) = B_g^{*i} + N_g^{*i} \cdot (\hat{X}_{0G}^i - X_{0G}) \qquad \text{Eq. 6.51}$$

Adding Eq. 6.51 for all the arcs yields:

$$\sum_{i=1}^{k}\left(N_g^{*i}\cdot(\hat{X}_{0G}-X_{0G})\right)=\sum_{i=1}^{k}\left(B_g^{*i}+N_g^{*i}\cdot(\hat{X}_{0G}^{i}-X_{0G})\right) \qquad \text{Eq. 6.52}$$

from which:

$$\hat{X}_{0G}-X_{0G}=\left(\sum_{i=1}^{k}N_g^{*i}\right)^{-1}\cdot\sum_{i=1}^{k}\left(B_g^{*i}+N_g^{*i}\cdot(\hat{X}_{0G}^{i}-X_{0G})\right) \qquad \text{Eq. 6.53}$$

where *k* is the number of arcs. A-priori information of the covariance of the global parameters can be introduced inEq. 6.53, which becomes:

$$\hat{X}_{0G}-X_{0G}=\left(P_{0G}^{-1}+\sum_{i=1}^{k}N_g^{*i}\right)^{-1}\cdot\sum_{i=1}^{k}\left(B_g^{*i}+N_g^{*i}\cdot(\hat{X}_{0G}^{i}-X_{0G})\right) \qquad \text{Eq. 6.54}$$

The practical application of the above formulae is the following: the different arcs are solved estimating both arc and common parameters and matrices N_g^{*i}, B_g^{*i} and the estimate of the common parameters $\hat{X}_{0G}^{i}$ are stored. New a-priori values X_{0G} are selected for the common parameters and contributions to the right-hand side of Eq. 6.54 are evaluated for each arc. After that Eq. 6.54 is solved to obtain the final estimate of the global parameters, $\hat{X}_{0G}$. The a-posteriori covariance matrix of these parameters is of course, from Eq. 6.54:

$$\hat{P}_{0G}^{-1}=\left(P_{0G}^{-1}+\sum_{i=1}^{k}N_g^{*i}\right)^{-1}$$

6.3.3 Method 3 (NEW MULTIARC, normal equation stacking)

The normal equation handling and stacking methods are based on the theory given in RD-35.

Both arc and common parameters are estimated on each arc. The normal equations for each arc *i* can be expressed as:

$$\begin{bmatrix} N_a^i & N_{aG}^i \\ N_{aG}^{iT} & N_G^i \end{bmatrix}\cdot\begin{bmatrix} \hat{X}_a^{i*}-\hat{X}_a^i \\ \hat{X}_G^{i*}-\hat{X}_G^i \end{bmatrix}=\begin{bmatrix} B_a^i \\ B_G^i \end{bmatrix} \qquad \text{Eq. 6.55}$$

where:

$$N_a^i = P_{ai}^{-1} + F_a^{iT} \cdot Q_{0i}^{-1} \cdot F_a^i$$

$$N_{aG}^i = F_a^{iT} \cdot Q_{0i}^{-1} \cdot F_G^i$$

$$N_G^i = P_{Gi}^{-1} + F_G^{iT} \cdot Q_{0i}^{-1} \cdot F_G^i$$

$$B_a^i = F_a^{iT} \cdot Q_{0i}^{-1} \cdot \Delta Y_i + P_{ai}^{-1}(X_{0a}^i - \hat{X}_a^i)$$

$$B_g^i = F_g^{iT} \cdot Q_{0i}^{-1} \cdot \Delta Y_i + P_{Gi}^{-1}(X_{0G}^i - \hat{X}_G^i)$$

Algorithm:

$$\begin{bmatrix} N_a^i & N_{aG}^i \\ N_{aG}^{iT} & N_G^i \end{bmatrix} \cdot \begin{bmatrix} \hat{X}_a^{i*} - \hat{X}_a^i \\ \hat{X}_G^{i*} - X_{0G} + X_{0G} - \hat{X}_G^i \end{bmatrix} = \begin{bmatrix} B_a^i \\ B_G^i \end{bmatrix} \qquad \text{Eq. 6.56}$$

whence:

$$\hat{X}_G^* - X_{0G} = \left[P_G^{-1} + \sum_{i=1}^{n} \left(N_G^i - N_{aG}^{iT} \cdot (N_a^i)^{-1} \cdot N_{aG}^i \right) \right]^{-1} \cdot \left[\sum_{i=1}^{n} \left(C_G^i - N_{aG}^{iT} \cdot (N_a^i)^{-1} \cdot C_a^i \right) \right]$$

$$\hat{X}_a^{i*} - \hat{X}_a^i = (N_a^i)^{-1} \cdot \left(C_a^i - N_{aG}^i \cdot (\hat{X}_G^* - X_{0G}) \right), \quad \forall i = 1, \ldots, n$$

where:

$$C_a^i = B_a^i + N_{aG}^i (\hat{X}_G^i - X_{0G}), \quad i = 1, \ldots, n$$

$$C_G^i = B_G^i + N_G^i (\hat{X}_G^i - X_{0G}), \quad i = 1, \ldots, n$$

New covariances:

$$P_G^* = \left[P_G^{-1} + \sum_{i=1}^{n} \left(N_G^i - N_{aG}^{iT} \cdot (N_a^i)^{-1} \cdot N_{aG}^i \right) \right]^{-1}$$

$$P_{ai}^* = (N_a^i)^{-1} + \left((N_a^i)^{-1} \cdot N_{aG}^i \right) \cdot P_G^{-1} \cdot \left(N_{aG}^{iT} \cdot (N_a^i)^{-1} \right), \quad i = 1, \ldots, n$$

New residuals:

$$\Delta Y_i^* = \Delta Y_i - F_a^i \cdot (\hat{X}_a^{i*} - \hat{X}_a^i) - F_G^i \cdot (\hat{X}_G^* - \hat{X}_G^i)$$

6.3.4 Parameter preelimination

The method of pre-elimination of parameters is a basic tool to reduce the dimension of the normal equation system without loosing information. Only the information about the pre-eliminated parameters is lost. With a separation of the parameter vector **x** into the vectors **x₁** and **x₂** we may write the NEQ system in the following form:

$$\begin{bmatrix} \mathbf{N}_{11} & \mathbf{N}_{12} \\ \mathbf{N}_{21} & \mathbf{N}_{22} \end{bmatrix} \begin{bmatrix} \mathbf{x}_1 \\ \mathbf{x}_2 \end{bmatrix} = \begin{bmatrix} \mathbf{z}_1 \\ \mathbf{z}_2 \end{bmatrix} \text{ to eliminate } \mathbf{z}_2 \text{ the second line is multiplied by } -\mathbf{N}_{12}\mathbf{N}_{22}^{-1}\mathbf{x}_2 \text{ this gives :}$$

$$\begin{bmatrix} \mathbf{N}_{11} & \mathbf{N}_{12} \\ -\mathbf{N}_{12}\mathbf{N}_{22}^{-1}\mathbf{N}_{21} & -\mathbf{N}_{12} \end{bmatrix} \begin{bmatrix} \mathbf{x}_1 \\ \mathbf{x}_2 \end{bmatrix} = \begin{bmatrix} \mathbf{z}_1 \\ -\mathbf{N}_{12}\mathbf{N}_{22}^{-1}\mathbf{z}_2 \end{bmatrix}$$

Evaluating the matrix multiplication and adding the resulting two equations gives:

$$\left(\mathbf{N}_{11} - \mathbf{N}_{12}\mathbf{N}_{22}^{-1}\mathbf{N}_{21}\right)\mathbf{x}_1 = \mathbf{z}_1 - \mathbf{N}_{12}\mathbf{N}_{22}^{-1}\mathbf{z}_2 \quad \text{or short} \quad \tilde{\mathbf{N}}_1\mathbf{x}_1 = \tilde{\mathbf{z}}_1$$

This new NEQ is reduced by the parameter vector **x₂**. However, thanks to the correction terms the resulting NEQ still contains the full information coming from the pre-eliminated parameters **x₂**.

6.3.5 Parameter transformation within normal equations

The transformation of parameters is especially important for the stacking of consecutive orbit arcs (and also used in ambiguity fixing methods). Its basic theory is as follows:

Starting from the normal equation expressed as:

$$\mathbf{N}\mathbf{x} = \mathbf{z}$$

Transforming this normal equation set to a new set of estimates $\tilde{\mathbf{x}}$ (or also $\Delta\tilde{\mathbf{x}}$). The relation of the original estimates and the new estimates can be written as:

$$\mathbf{x} = \mathbf{B}\tilde{\mathbf{x}} + \mathbf{c}$$

Where:

$\mathbf{x}$ The original estimates.

$\tilde{\mathbf{x}}$ The transformed estimates.

$\mathbf{B}$ The n x n transformation matrix with n rows and n columns.

$\mathbf{c}$ A n x 1 vector of constants with n rows.

The transformed normal equation matrix can then be written as:

$$\tilde{\mathbf{N}}\tilde{\mathbf{x}} = \tilde{\mathbf{z}}$$

Where:

$\tilde{\mathbf{N}} = \mathbf{B}^{\mathbf{T}}\mathbf{NB}$ The transformed normal matrix.

$\tilde{\mathbf{z}} = \mathbf{B}^{\mathbf{T}}(\mathbf{z} - \mathbf{Nc})$ The transformed RHS (Right Hand Side).

6.3.6 Combination of consecutive orbital arcs

In order to stack consecutive NEQ systems in which the estimated satellite orbit parameters are made continuous at the NEQ boundaries (i.e. from various sets of consecutive state vectors, we get just one at the beginning of the first arc, but based on the information of all the arcs) the theory of parameter transformation is used.

The transformation matrix (**B**) and the vector of constants (**c**) are derived by requesting that the satellite positions from both NEQ systems are identical for a certain time t. This may be written as:

$$\mathbf{y}(t, \mathbf{r}_1, \mathbf{q}_1) = \mathbf{y}(t, \mathbf{r}_2, \mathbf{q}_2)$$

For continuity of the satellite orbits the dynamical satellite parameters, e.g., solar radiation and drag, have to be identical as well:

$$\mathbf{q}_1 = \mathbf{q}_2$$

Where:

$\mathbf{y}(t, \mathbf{r}_i, \mathbf{q}_i)$ Satellite position function (e.g., orbital model) as function of time and the a priori orbit parameters.

t The epoch for which the continuity of the state vector is requested

$\mathbf{r}_\mathrm{i}$ The state vector of the satellite of NEQ system i given for a time t_i

$\mathbf{q}_\mathrm{i}$ The dynamical orbit parameters of NEQ system i.

The linearised observation equation for the satellite positions may be written as:

$$\mathbf{y}(t,\mathbf{r}_\mathrm{i},\mathbf{q}_\mathrm{i}) = \mathbf{y}(t,\mathbf{r}_{\mathrm{i}_0},\mathbf{q}_{\mathrm{i}_0}) + \frac{\partial \mathbf{y}(t,\mathbf{r}_{\mathrm{i}_0},\mathbf{q}_{\mathrm{i}_0})}{\partial \mathbf{r}}\Delta\mathbf{r}_\mathrm{i} + \frac{\partial \mathbf{y}(t,\mathbf{r}_{\mathrm{i}_0},\mathbf{q}_{\mathrm{i}_0})}{\partial \mathbf{q}}\Delta\mathbf{q}_\mathrm{i}$$

The linearised observation equation for the dynamical parameters may be written as:

$$\mathbf{q}_\mathrm{i} = \mathbf{q}_{\mathrm{i}_0} + \Delta\mathbf{q}_\mathrm{i}$$

If we demand continuity of the estimated orbital parameters from two NEQ files, as given above, we can now write this as:

$$\mathbf{y}(t,\mathbf{r}_{1_0},\mathbf{q}_{1_0}) + \frac{\partial \mathbf{y}(t,\mathbf{r}_{1_0},\mathbf{q}_{1_0})}{\partial \mathbf{r}}\Delta\mathbf{r}_1 + \frac{\partial \mathbf{y}(t,\mathbf{r}_{1_0},\mathbf{q}_{1_0})}{\partial \mathbf{q}}\Delta\mathbf{q}_1 = \mathbf{y}(t,\mathbf{r}_{2_0},\mathbf{q}_{2_0}) + \frac{\partial \mathbf{y}(t,\mathbf{r}_{2_0},\mathbf{q}_{2_0})}{\partial \mathbf{r}}\Delta\mathbf{r}_2 + \frac{\partial \mathbf{y}(t,\mathbf{r}_{2_0},\mathbf{q}_{2_0})}{\partial \mathbf{q}}\Delta\mathbf{q}_2$$

And for the dynamical parameters as:

$$\mathbf{q}_{1_0} + \Delta\mathbf{q}_1 = \mathbf{q}_{2_0} + \Delta\mathbf{q}_2$$

Putting both equations together we can write this in matrix form as:

$$\begin{bmatrix} \frac{\partial \mathbf{y}(t,\mathbf{r}_{2_0},\mathbf{q}_{2_0})}{\partial \mathbf{r}} & \frac{\partial \mathbf{y}(t,\mathbf{r}_{2_0},\mathbf{q}_{2_0})}{\partial \mathbf{q}} \\ \mathbf{0} & \mathbf{1} \end{bmatrix} \begin{bmatrix} \Delta\mathbf{r}_2 \\ \Delta\mathbf{q}_2 \end{bmatrix} = \begin{bmatrix} \frac{\partial \mathbf{y}(t,\mathbf{r}_{1_0},\mathbf{q}_{1_0})}{\partial \mathbf{r}} & \frac{\partial \mathbf{y}(t,\mathbf{r}_{1_0},\mathbf{q}_{1_0})}{\partial \mathbf{q}} \\ \mathbf{0} & \mathbf{1} \end{bmatrix} \begin{bmatrix} \Delta\mathbf{r}_1 \\ \Delta\mathbf{q}_1 \end{bmatrix} + \begin{bmatrix} \mathbf{y}(t,\mathbf{r}_{1_0},\mathbf{q}_{1_0}) - \mathbf{y}(t,\mathbf{r}_{2_0},\mathbf{q}_{2_0}) \\ \mathbf{q}_{1_0} - \mathbf{q}_{2_0} \end{bmatrix}$$

This may be written in the following simplified form:

$$\mathbf{Ox} = \mathbf{P}\tilde{\mathbf{x}} + \mathbf{c}$$

If we want to write this in the form of the transformation equation $\mathbf{x} = \mathbf{B}\tilde{\mathbf{x}} + \mathbf{c}$ we get:

$$\mathbf{x} = \mathbf{O}^{-1}\mathbf{P}\tilde{\mathbf{x}} + \mathbf{O}^{-1}\mathbf{c}$$

Where:

$$\mathbf{O} = \begin{bmatrix} \frac{\partial \mathbf{y}(t,\mathbf{r}_{2_0},\mathbf{q}_{2_0})}{\partial \mathbf{r}} & \frac{\partial \mathbf{y}(t,\mathbf{r}_{2_0},\mathbf{q}_{2_0})}{\partial \mathbf{q}} \\ \mathbf{0} & \mathbf{1} \end{bmatrix}$$

$$\mathbf{P} = \begin{bmatrix} \frac{\partial \mathbf{y}(t,\mathbf{r}_{1_0},\mathbf{q}_{1_0})}{\partial \mathbf{r}} & \frac{\partial \mathbf{y}(t,\mathbf{r}_{1_0},\mathbf{q}_{1_0})}{\partial \mathbf{q}} \\ \mathbf{0} & \mathbf{1} \end{bmatrix}$$

$$\mathbf{c} = \begin{bmatrix} \mathbf{y}(t,\mathbf{r}_{1_0},\mathbf{q}_{1_0}) - \mathbf{y}(t,\mathbf{r}_{2_0},\mathbf{q}_{2_0}) \\ \mathbf{q}_{1_0} - \mathbf{q}_{2_0} \end{bmatrix}$$

So to be able to combine the orbital estimates from two different NEQ systems to result in one unique and continuous set of estimates over the time interval covered by the two NEQ systems we need to the following information:

$\mathbf{r}_{\mathbf{i}_0},\mathbf{q}_{\mathbf{i}_0}$ The a priori state vectors and dynamical parameters used for generating the different NEQ systems.

$\mathbf{y}(t,\mathbf{r}_{\mathbf{i}_0},\mathbf{q}_{\mathbf{i}_0})$ The position and velocity vector of the satellites at the selected point in time (t) where we enforce the continuity of the orbits.

$\dfrac{\partial\mathbf{y}(t,\mathbf{r}_{\mathbf{i}_0},\mathbf{q}_{\mathbf{i}_0})}{\partial\mathbf{r}}$ The partial derivatives with respect to the satellite state vector at the selected point in time (t)

$\dfrac{\partial\mathbf{y}(t,\mathbf{r}_{\mathbf{i}_0},\mathbf{q}_{\mathbf{i}_0})}{\partial\mathbf{q}}$ The partial derivatives with respect to the dynamical satellite parameters at the selected point in time (t)

To enable this orbit combination in Napeos the saved normal equation systems must contain this information. The main assumption in this case is that the orbit combination will always be based on combining consecutive NEQ systems. It should therefore be sufficient to save the above information only for the very first and the very last epoch of the NEQ. However, the last epoch of a NEQ should be the same as the first epoch of the next consecutive NEQ.

6.3.7 Converting pass parameters in arc parameters

The starting point is the already computed normal equations

$$\begin{bmatrix} N_1 & 0 & \cdots & 0 & N_{1A} \\ 0 & N_2 & \cdots & 0 & N_{2A} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \cdots & N_p & N_{pA} \\ N_{1A}^T & N_{2A}^T & \cdots & N_{pA}^T & N_A \end{bmatrix} \cdot \begin{bmatrix} \hat{X}_1^* - \hat{X}_1 \\ \hat{X}_2^* - \hat{X}_2 \\ \vdots \\ \hat{X}_p^* - \hat{X}_p \\ \hat{X}_A^* - \hat{X}_A \end{bmatrix} = \begin{bmatrix} B_1 \\ B_2 \\ \vdots \\ B_p \\ B_A \end{bmatrix} \qquad \text{Eq. 6.57}$$

where:

$$N_i = P_i^{-1} + F_i^T \cdot Q_i^{-1} \cdot F_i, \quad i = 1,\dots,p$$

$$N_{iA} = F_i^T \cdot Q_i^{-1} \cdot F_A, \quad i = 1,\dots,$$

$$N_i = P_A^{-1} + \sum_{i=1}^{p} \left(F_{iA}^T \cdot Q_i^{-1} \cdot F_{iA} \right) + F_A^T \cdot Q_A^{-1} \cdot F_A$$

$$B_i = F_i^T \cdot Q_i^{-1} \cdot \Delta Y_i + P_i^{-1}(X_{0i} - \hat{X}_i), \quad i = 1,\dots,p$$

$$B_A = \sum_{i=1}^{p} \left(F_{iA}^T \cdot Q_i^{-1} \cdot \Delta Y_i \right) + F_A^T \cdot Q_A^{-1} \cdot \Delta Y_A + P_A^{-1}(X_{0A} - \hat{X}_A)$$

and the $N_i = (n_{j,k})$ are the subsections containing the partial derivatives with respect to the parameters estimated by pass (or time interval) and that are to be estimated by arc. And the summations on '*j*' represent the '*n*' elements not participating in the conversion.

The $\hat{X}_i^*$ represents for '*i*' a vector whose all identical components, $\hat{\beta}_i^*$, are the new arc estimate of the grouped parameter '*i*'.

Algorithm:

$$\begin{bmatrix} N_1 & 0 & \cdots & 0 & N_{1A} \\ 0 & N_2 & \cdots & 0 & N_{2A} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \cdots & N_p & N_{pA} \\ N_{1A}^T & N_{2A}^T & \cdots & N_{pA}^T & N_A \end{bmatrix} \cdot \begin{bmatrix} \hat{X}_1^* - X_1 + X_1 - \hat{X}_1 \\ \hat{X}_2^* - X_2 + X_2 - \hat{X}_2 \\ \vdots \\ \hat{X}_p^* - X_p + X_p - \hat{X}_p \\ \hat{X}_A^* - \hat{X}_A \end{bmatrix} = \begin{bmatrix} B_1 \\ B_2 \\ \vdots \\ B_p \\ B_A \end{bmatrix} \qquad \text{Eq. 6.58}$$

Whence:

$$N_i \cdot (\hat{X}_i^* - X_i) + N_A \cdot (\hat{X}_A^* - \hat{X}_A) = C_i, \quad i = 1,\dots,p$$

$$\sum_{i=1}^{p} \left(N_{iA}^T \cdot (\hat{X}_i^* - X_i) + N_A \cdot (\hat{X}_A^* - \hat{X}_A) \right) = C_A, \quad i = 1,\dots,p$$

$$C_i = B_i - N_i \cdot (X_i - \hat{X}_i), \quad i = 1,\dots,p$$

$$C_A = B_A - \sum_{i=1}^{p} \left(N_{iA}^T \cdot (X_i - \hat{X}_i) \right)$$

Expanding the previous equation in all their components and taking into account that $\hat{X}_i^*$ and X_i have all their elements identical (in $n_{j,k}$ first index denotes column)

$$\left(\hat{\beta}_i^* - \beta_i\right)\cdot\left(\sum_{j=1}^{m_j} n_{j,k}\right)_i + \left(\sum_{j=1}^{m_A}\left(\left((\hat{x}_j)_A^* - (\hat{x}_j)_A\right)\cdot(n_{j,k})_{iA}\right)\right)_i = (c_k)_i \quad , \quad \begin{cases} i = 1,\ldots,p \\ k = 1,\ldots,m_i \end{cases}$$

$$(c_k)_i = (b_k)_i - \left(\sum_{j=1}^{m_j}\left(\beta_i\cdot(\hat{x}_j)_A\right)n_{j,k}\right)_i \quad , \quad \begin{cases} i = 1,\ldots,p \\ k = 1,\ldots,m_i \end{cases}$$

$$\sum_{i=1}^{p}\left(\left(\hat{\beta}_i^* - \beta_i\right)\left(\sum_{j=1}^{m_j} n_{j,k}\right)_{i,A}\right) + \left(\sum_{j=1}^{m_A}\left(\left((\hat{x}_j)_A^* - (\hat{x}_j)_A\right)\cdot(n_{j,k})_{iA}\right)\right)_i = (c_k)_A \quad , \quad k = 1,\ldots,m_A \qquad \text{Eq. 6.59}$$

$$(c_k)_A = (b_k)_A - \sum_{i=1}^{p}\left(\sum_{j=1}^{m_j}\left(\beta_i\cdot(\hat{x}_j)_A\right)n_{j,k}\right)_i \quad , \quad k = 1,\ldots,m_A$$

$$\beta_i = \frac{1}{m_i}\sum_{j=1}^{m_i}(x_j)_i \quad , \quad i = 1,\ldots,p$$

Whence, adding all equations for the same parameter

$$\left(\hat{\beta}_i^* - \beta_i\right)\cdot\sum_{j=1}^{m_j}\sum_{k=1}^{m_j}\left(n_{j,k}\right) + \left(\sum_{j=1}^{m_A}\left(\left((\hat{x}_j)_A^* - (\hat{x}_j)_A\right)\cdot\sum_{k=1}^{m_j}\left(n_{j,k}\right)_A\right)\right)_i = \left(\sum_{k=1}^{m_j}(c_k)\right)_i \quad , \quad i = 1,\ldots,p \qquad \text{Eq. 6.60}$$

The final common value for each parameter being grouped is

$$\left(\hat{\beta}_i^* - \beta_i\right) = \frac{\left(\sum_{k=1}^{m_j}(c_k)\right)_i - \sum_{j=1}^{m_A}\left(\left((\hat{x}_j)_A^* - (\hat{x}_j)_A\right)\cdot\sum_{k=1}^{m_j}\left(n_{j,k}\right)_A\right)}{\sum_{j=1}^{m_j}\sum_{k=1}^{m_j}\left(n_{j,k}\right)} \quad , \quad i = 1,\ldots,p \qquad \text{Eq. 6.61}$$

The resulting new values for the parameters not being re-estimated are

$$\left(\hat{X}_A{}^* - \hat{X}_A\right) = \hat{X}_A{}^{-1}\left(C_A - \sum_{i=1}^{p}\left((N_{iA}{}^*)^T \cdot (\hat{\beta}_i^* - \beta_i)\right)\right)$$

$$N_{iA}{}^* = \sum_{k=1}^{m_j} n_{j,k} \quad , \quad \begin{cases} i = 1,\ldots,p \\ k = 1,\ldots,m_A \end{cases} \qquad \text{Eq. 6.62}$$

6.3.8 Ground Station Velocity Estimation

The theory of parameter transformation shall be used to introduce the station velocities. For this purpose the position of the station shall be represented as an initial position plus a velocity times the time. Since we are introducing a new type of parameter, a minimum of two consecutive NEQs to avoid singularity is required (this should not be a problem since it is expected that the NEQs to be processed shall correspond to one or two years). It is assumed that the influence of the station velocity in the sequential solution is negligible. Mathematically, starting from a system defined by:

$$Nx = b \equiv \begin{bmatrix} N_{12}N_{12} \\ N_{21}N_{22} \end{bmatrix}\begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \end{bmatrix} \qquad \text{Eq. 6.63}$$

where N represents the normal equations, x the parameters to estimate and b the right hand side of the system. Subindex 1 represents all the parameters except the station positions, and subindex 2 corresponds to the station positions.

Representing the position of the stations as:

$$x_2 = x_{20} + v_2 \Delta t$$

Being Δt the time span covered by the individual NEQ.

The resulting system of equations is:

$$\begin{bmatrix} N_{11} & N_{12} & \Delta t N_{12} \\ N_{21} & N_{22} & \Delta t N_{22} \\ \Delta t N_{21} & \Delta t N_{22} & \Delta t^2 N_{22} \end{bmatrix}\begin{bmatrix} x_1 \\ x_{20} \\ v_2 \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \\ \Delta t b_2 \end{bmatrix} \qquad \text{Eq. 6.64}$$

So transforming the NEQ to this system (and ensuring same reference position for the initial position of the station in the consecutive equations), the sequential NEQs are stacked.

6.3.9 Fixed Orbit

In the above sections it was always assumed that the satellite initial state vector (position and velocity) is being estimated and propagated at every iteration, together with the variational partials.

However, if the satellite orbit (position and velocity) is already available it is possible to use all the methods described above in the same way. Only non-dynamic parameters can be estimated in this case because the estimation of dynamic parameters (satellite state vector and force model parameters) requires additionally the evaluation of variational partials.

In this case the observation equation coefficients of matrix F and the computed observations are calculated using the satellite position and velocity from the available orbit and there is no need to propagate the orbit every new iteration. Note however that the solution of normal equations is still iterative because matrix F and the residuals ΔY depend also on the estimated parameters themselves.

6.4 Sequential Estimation

In sequential estimation observations are processed one by one or in small groups as soon as they are received and estimates of the satellite state vector and parameters at the observation time are evaluated. One advantage of this approach is that the orbit can be calculated in near-real time. Another advantage is that the dimension of the matrix to be inverted will be equal to the number of observations processed: if observations are processed individually only scalar divisions will be required. Sequential estimation methods are described in RD-22, RD-26, and RD-31. For sequential estimation normal equation stacking may be used but a Kalman filter might be better suited, depending on the application.

6.4.1 Extended Kalman Filter

In the problem of batch estimation described in section 6.2, the estimate of the satellite state vector at a time t later than the epoch t_0 is obtained by propagating the value of $\hat{X}_0$ obtained in the last iteration. The Kalman filter on the other hand estimates the state vector at t directly based on all observations up to this time

Let's assume now that at some time $t1$ an optimal estimate of the state vector (extended) $\hat{X}_1$ and its covariance matrix $\hat{P}_1$ are known, and a set of q observations taken at a later time $t2$ is received, their covariance matrix being Q_{02}. The extended Kalman filter uses the following equations to evaluate the best estimate of $\hat{X}_2$ and $\hat{P}_2$:

$$\hat{X}_2 = \hat{X}_2^* + K_2 \cdot \Delta Y_2$$
$$\hat{P}_2^* = \Phi(t_2,t_1) \cdot \hat{P}_1 \cdot \Phi(t_2,t_1)^T \qquad \text{Eq. 6.65}$$
$$\hat{P}_2 = (I - K_2 \cdot F_2) \cdot \hat{P}_2^*$$

where $\hat{X}_2^*$ is the propagation of $\hat{X}_1$, ΔY_2 is the residual matrix of the observations at t_2, F_2 is the matrix of observation coefficients of the observations at t_2. ΔY_2 and F_2 are calculated using $\hat{X}_2^*$. Note that F_2 are the derivatives of the observations at t_2 w.r.t. the parameters at t_2, therefore no variational partials are required in their evaluation. Matrix K_2 is the ***gain matrix*** and it is given by:

$$K_2 = \hat{P}_2^* \cdot F_2^T \cdot \left(Q_{02} + F_2 \cdot \hat{P}_2^* \cdot F_2^T\right)^{-1} \qquad \text{Eq. 6.66}$$

$\Phi(t_2,t_1)$ is the transition matrix of the parameters at t_2 w.r.t. the parameters at t_1, and it is similar to the one defined in Eq. 6.23 but extended to all the parameters. If we partition the vector of estimated parameters in satellite position/velocity, dynamic parameters and non-dynamic parameters, matrix Φ can be expressed as:

$$\Phi = \begin{bmatrix} \phi & \psi & 0 \\ 0 & I & 0 \\ 0 & 0 & I \end{bmatrix}$$

ϕ is the *(6x6)* transition matrix of the satellite state vector (position and velocity), as described in Eq. 6.23. ψ are the partial derivatives of the satellite state vector w.r.t. the dynamical parameters (except of course the satellite state vector itself which is already considered in ϕ). Take into account that:

$$[\phi \quad \psi] = \begin{bmatrix} X_m \\ \dot{X}_m \end{bmatrix}$$

That is, matrix Φ can be evaluated by integration of the variational equations from t_1 to t_2.

Eq. 6.65 can be used recursively to estimate $\hat{X}_1, \hat{P}_1, \hat{X}_2, \hat{P}_2, \hat{X}_3, \hat{P}_3, \ldots$. $\hat{X}^*_{k+1}$ is propagated from $\hat{X}_k$ and then improved with observations using first equation in Eq. 6.65. $\hat{P}^*_{k+1}$ is propagated from $\hat{P}_k$ using second equation in Eq. 6.65 and then improved with observations using third equation in Eq. 6.65. There are no iterations involved in this process, unlike batch estimation. A disadvantage of this method is that propagation has to be restarted after each observation or set of observations has been processed.

Note that the matrix to be inverted in Eq. 6.66 is of dimension *(qxq)*. In the case *q=1* the filter equations are significantly simplified since the inversion of the matrix needed in Eq. 6.66 to compute the gain matrix is reduced to a scalar inversion. If it can be shown that the result of processing the *q* observations simultaneously is the same as if these observations are sequentially processed.

Because of computer round-off errors, $\hat{P}_2$ as calculated in third equation in Eq. 6.65 can become non-positive definite and therefore meaningless. An alternative is to use the *Joseph algorithm* for its computation:

$$\hat{P}_2 = (I - K_2 \cdot F_2) \cdot \hat{P}_2^* \cdot (I - K_2 \cdot F_2)^T + K_2 \cdot Q_{02} \cdot K_2^T \qquad \text{Eq. 6.67}$$

Eq. 6.67 requires more computation than third equation in Eq. 6.65, but ensures that $\hat{P}_2$ will remain positive definite.

Due to computer round-off, matrix will tend to zero as more observations are processed. This in turn will cause the gain matrix to approach zero (see Eq. 6.66) and the estimation algorithm will become insensitive to any further observation, which will lead to filter divergence. One way of preventing the filter from diverging is to add a certain level of process noise to the system model to account for the model error contributions. Second equation in Eq. 6.65 becomes then:

$$\hat{P}_2^* = \Phi(t_2, t_1) \cdot \hat{P}_1 \cdot \Phi(t_2, t_1)^T + R_2 \qquad \text{Eq. 6.68}$$

where R_2 represents the process noise. The addition of process noise prevents the covariance matrix from approaching zero, therefore preventing the gain matrix from approaching zero. The choice of the appropriate level of process noise is largely heuristic and depends to a large extent on what is known about the unmodelled state parameters.

An alternative to these sometimes ad hoc stabilization techniques is to modify or replace the algorithm by one that is mathematically equivalent but numerically better conditioned. The *square root information filter (SRIF)* is such a solution.

7 EARTH OBSERVATION

7.1 Introduction

For Earth Observation missions, the low polar near circular orbit is the most popular used. The major advantage for such an orbit is that the combination of the Earth's rotation about its axis and the spacecraft's orbital motion around the Earth results in very complete scanning of the globe. Moreover low polar orbit allows a close inspection of any point on the Earth's surface as described in RD-1.

For most of payload instruments mounted on the spacecraft to observe the Earth, the point on the Earth's surface that is observed depends only on the position of the spacecraft or on its position and direction of travel. To ensure that the Earth's surface is observed in a regular and systematic manner, a certain **repeat pattern** has to be maintained, such that the spacecraft follows the same path relative to the Earth and therefore the same point can be observed several times. As a first approximation, for the spacecraft to be in a periodic orbit relative to the rotating Earth, the spacecraft orbital period (P) must be a rational fraction n/m of the Earth's spin period (Ψ), such that after m spacecraft orbits and n Earth rotations, the spacecraft will be over the same point and its motion relative to the Earth will be repeated.

However it is known that some perturbations are acting on the spacecraft such that this repeated pattern can be affected. For low polar orbits major perturbations to take into account can be stated as the perturbations caused by the Earth Gravity Field, caused by the attraction of the Sun and Moon, and the air drag.

Due to the Earth Gravity Field, mainly the Earth's oblateness term, there are some perturbations acting on the spacecraft orbit. These are:

- The precession of the orbital plane. This can be selected to be equal to the mean angular rate of the Earth in its orbit around the Sun. Such an orbit is called **Sun-synchronous**. The advantage of this Sun-synchronisation for Earth observation satellites using optical instruments is that the lighting conditions for the surface being overflown remain comparatively constant. For other satellites without these optical instruments, the advantage lies in the spacecraft solar panels that need only be rotated around one axis to maintain pointing of the Sun's direction.

  In order to have a repeated pattern, the discussion above is valid, but the orbital period (P) must be interpreted as nodal period (i.e. time between two consecutive crossings of the equatorial plane from south to north) and the Earth's spin period (Ψ), must be interpreted as the spin rate of the Earth relative to the precessing orbital plane.
- A periodic perturbation repeating each orbital revolution, which will generate a cumulative deformation of the orbit shape. For a near circular orbit, what is intended is that the spacecraft has equal altitudes for consecutive passes over the same area. After a time, the deformed orbit would have essentially the same orbital period, but the altitude

shall have varied, and therefore the spacecraft velocity would be less constant, increasing in parts with reduced altitude and decreasing where the altitude has increased.

The amount of deformation will be quite sensitive to the precise shape of the initial orbit. It is possible in practise to select an initial orbit such that this deformation is essentially zero. Such an orbit is called **frozen orbit**. With an optimal frozen orbit, all perturbations associated with the Earth's gravitational field will essentially be periodic, and therefore after *n* days and *m* nodal orbits the path relative to the Earth is repeated just the same.

Due to the gravitational forces of the Sun and Moon, the orbit's mean inclination will decrease slowly. Therefore inclination correction manoeuvres are needed to compensate for these reductions.

Due to air drag, the spacecraft's orbital velocity will be reduced, and therefore its altitude and the nodal period. This drag is strongly dependant on the solar activity, which is difficult to be predicted. Therefore regular orbit maintenance manoeuvres are necessary in order to compensate the air drag.

7.2 The Reference Orbit. Target Computation

As derived from the discussed above, in absence of air drag and Sun and Moon gravitational perturbations, it is possible for a spacecraft in low polar near circular orbit to have almost repetitive orbits and therefore a repeat pattern. Such theoretical orbit is referred to as the **reference orbit** and shall be a Sun-synchronous, frozen orbit, with a nodal period equal to a rational fraction *n/m* of the Earth's spin period relative to the precessing orbital plane. This reference orbit, expressed in a fixed geodetic coordinate system will define the **reference ground track**.

However accounting for air drag and Sun and Moon gravitational perturbations this reference ground track can not be followed by the spacecraft unless considering certain tolerances. The maximum deviation allowed from this reference ground track defines the **deadband**. For Earth Observation missions the main goal and therefore the target, is to maintain the spacecraft within this deadband. Therefore the computation of the **reference ground track** is the major step to define the target for this kind of mission.

7.2.1 Repeat Pattern

As discussed above, a first approximation to a repeat pattern could be to consider the spacecraft orbital period (P) as a rational fraction n/m of the Earth's spin period (Ψ). Therefore an approximate vale of the semi-major axis (a) of the reference orbit is directly derived.

$$P = \frac{n}{m}\Psi \text{ where } P = 2\pi\sqrt{\frac{a^3}{\mu}} \text{ and } \Psi = 86400.0\,seg \qquad \text{Eq. 7.1}$$

7.2.2 Sun-Synchronous Orbit

The precession rate of the orbital plane can be approximately computed, accounting for the dominant oblateness parameter J_2 with the following expression (RD-12).

$$\dot{\Omega} = \frac{d\Omega}{dt} = -\frac{3}{2}J_2\sqrt{\frac{\mu}{a^3}}\left(\frac{R}{a(1-e^2)}\right)^2 \cos i \qquad \text{Eq. 7.2}$$

where

Ω is the right ascension of the ascending node

e is the orbital eccentricity

i is the orbital inclination

R is the mean equatorial Earth radius

Sun-synchronous orbits are those with a secular rate of the right ascension of the ascending node equal to the right ascension rate of the mean sun:

$$\dot{\Omega}_{\text{SUN}} = 0.9856\,{}^{\text{deg}}\!/_{\text{day}} \qquad \text{Eq. 7.3}$$

Therefore for a satellite orbit to be sun-synchronous, the inclination i, semi-major axis a and eccentricity e must satisfy the above relation. For near circular orbits, the eccentricity vanishes (e = 0), and since the semi-major axis a is known from the period, using equations Eq. 7.1, Eq. 7.2 and Eq. 7.3, an approximate value of the inclination i can be derived.

7.2.3 Local Time at Node Crossing

The local time of the descending node is defined as the UTC time of the node passage corrected with the fraction of a day that corresponds to the angle between the node and the Greenwich meridian. The origin of the UTC time has been selected such that the Sun is close to zenith at longitude 180^{o} every midnight (00:00 Z). Therefore an approximate direction to the Sun (*S*) (projection of the direction to the equatorial plane) is

$$S = GMT - 360^{o} \cdot t + 180^{o}$$

where t is UTC time measured in days and *GMT* is the Greenwich Meridian Time at the initial epoch *t*:

$$GMT = STD20 + OMT20 \cdot t + OMQ20 \cdot t^{2}$$

The local time of a point on the Earth's surface situated vertically beneath the spacecraft is the dihedral angle between the meridian plane containing the spacecraft and the meridian plane containing the Earth-Sun direction. According to this definition the local time at ascending node LTA would be $LTA = \Omega - S$ and at descending node $LTD = LTA + 180^{o}$. This angle is equivalent to an hour using the following rule ($360 \deg \leftrightarrow 24\,h$), and the convention $12\,h \equiv 0 \deg$.

Therefore to constraint the local time at ascending or descending node crossing is completely equivalent to fix the right ascension of the ascending node Ω.

7.2.4 Overfly an Earth Point

For some Earth Observation missions (e.g. ERS-1), the reference orbit can be constrained to overfly an Earth point. This point is defined by its geodetic coordinates (*Long*, *Lat*). Its argument of latitude (*Latar*) is derived as:

$$\sin Latar \cdot \sin i = \sin Lat$$

The projection of this angle in the equatorial plane (*Latarp*) is computed as:

$$\tan Latarp = \frac{\sin Latar \cdot \cos i}{\cos Latar}$$

The ascending node is now directly derived as:

$$\Omega = Long + Latarp + GMT \qquad \text{Eq. 7.4}$$

The initial true anomaly ν corresponding with the Earth point, can be derived if the argument of perigee ω is known as follows:

$$\nu = Latar - \omega \qquad \text{Eq. 7.5}$$

7.2.5 Frozen Eccentricity Vector

As demonstrated in RD-14 the necessary and sufficient condition for having constant altitudes for consecutive values of equal geocentric latitude is as the secular perturbation of the eccentricity e and argument of perigee ω are zero. This is the well known problem of having a frozen orbit.

The algorithm to calculate a frozen orbit is well described in references RD-13 and RD-14. Hereafter the major steps of the algorithm will be summarized, avoiding intermediate mathematical demonstrations.

Firstly it is necessary to introduce the concept "eccentricity vector". This is the vector with length equal to the eccentricity e pointing from the centre of the Earth to the perigee. With rectangular coordinates in the orbital plane defined by the node this vector is:

$$\vec{e} = \left(e_x, e_y\right) = e\left(\cos\omega, \sin\omega\right) \qquad \text{Eq. 7.6}$$

An initial approximate orbit is integrated for an integer number of complete repeat cycles, and the averages of the eccentricity vector over the first and the last repeat cycle are computed. Assuming that the average eccentricity vector over complete repeat cycles follows a circle (or rather a polygon as demonstrated in RD-14, the centre of this circle with respect to the initial mean eccentricity vector can be computed using the formula:

$$\delta e_1 = \frac{\Delta e_1 - \Delta e_2 \cot\left(\dfrac{At}{2e_0}\right)}{2} \qquad \text{Eq. 7.7}$$

$$\delta e_2 = \frac{\Delta e_2 + \Delta e_1 \cot\left(\dfrac{At}{2e_0}\right)}{2} \qquad \text{Eq. 7.8}$$

where:

$$A = -\frac{3\sqrt{\dfrac{\mu}{a^3}} J_3 R^3 \sin i}{2a^3}\left(1 - \frac{5}{4}(\sin i)^2\right)$$

e_0 is a reference eccentricity around which it is linearised. For this iterative algorithm it can be approximated by the mean eccentricity of the first repeat cycle

R is the mean radius of the equator

$\left(\Delta e_1, \Delta e_2\right)$ are the components of the difference between the average eccentricity vector over the last and the first repeat cycle

t is the time between the starts of the first and last repeat cycle

$(\delta e_1, \delta e_2)$ are the components of the difference between the centre of the circle and the average eccentricity vector over the first repeat cycle

Therefore, $(\delta e_1, \delta e_2)$ is the increment that should be made to the mean eccentricity vector over the first repeat cycle to have no secular perturbation of the eccentricity vector. For the algorithm, this increment is applied to the initial **osculating** eccentricity vector resulting in approximately the same increment in mean eccentricity vector and the corresponding initial state vector in rectangular coordinates required for the numerical integration can be computed using the standard formulas.

It is clear that if the mean eccentricity vector is constrained to a prescribed values (e_{xp}, e_{yp}), it is only necessary to propagate through the first repeat cycle, compute its mean eccentricity vector (e_{x1}, e_{y1}) and $(\delta e_1, \delta e_2)$ will become in:

$$(\delta e_1, \delta e_2) = (e_{xp}, e_{yp}) - (e_{x1}, e_{y1}) \qquad \text{Eq. 7.9}$$

7.2.6 Algorithm

Based on the discussions above, the final algorithm to compute the reference orbit and subsequently the reference ground track can be derived. It consists of the following steps:

- The user shall supply the initial guesses of the orbital elements at a time, for the reference orbit, either directly or indirectly, via the items mentioned previously. Summarising:
    - Repeat pattern from which the semi-major axis *a* is derived.
    - Sun-synchronous from which the inclination *i* is derived.
    - Local Time at node crossing from which the right ascension of the ascending node *Ω* is derived
    - Overflying an Earth point at the indicated time, from which the right ascension of the ascending node Ω and the true anomaly ν are derived
    - An initial estimate of the eccentricity e and the argument of perigee ω is necessary. If no better information is available they can be set to values indicated in RD-14:

      $$e = -\frac{J_3 R}{2 J_2 a} \sin i$$

      $$\omega = 90^{\circ}$$

      For orbits similar to ERS-1 and ERS-2 a better estimate can be selected as mentioned in RD-14:

$e = 0.001165$

$\omega = 90^{\circ}$

Notice that if the right ascension of the ascending node Ω is derived from the local time at node crossing, it is necessary to provide the initial guess of the true anomaly of the reference orbit.

- The initial estate vector shall be propagated for a number of repeat cycles as required for the frozen eccentricity vector algorithm. After the first repeat cycle, it is also obtained the Earth point overflown and the local time at node crossing. With this information the new orbital elements for the reference orbit can be adjusted as follows:
  - To repeat a pattern, the Earth point overflown at the starting time of two consecutive repeat cycle must be the same. If $\Delta\lambda$ is referred to the difference in longitude of the Earth point overflown at the starting time of two consecutive repeat cycle and corresponding to the same latitude, the semi-major axis a, can be modified in order to absorb this $\Delta\lambda$ as follows:

    $\Delta\lambda = \omega_E \Delta P \frac{\Delta T}{P}$ and considering that $\frac{\Delta P}{P} = \frac{3}{2}\frac{\Delta a}{a}$ results in:

    $$\Delta a = \frac{a\Delta\lambda}{\frac{3}{2}\omega_E \Delta T}$$

    where

    ω_E is the Earth's spin

    P is the s/c orbital period

    ΔT is the elapsed time
  - To maintain constant the local time at node crossing the inclination i can be adjusted as follows:

    $$\Delta LTD = \Delta\Omega = \frac{\partial\Omega}{\partial i} N\Delta i \quad \Rightarrow \quad \Delta i = \frac{\Delta\Omega}{N\frac{3}{2}2\pi R_{\oplus}{}^{2}\frac{J_2}{a^2}\sin i} \qquad \text{Eq. 7.10}$$

    where

    ΔLTD is the difference in the local time at node crossing at starting of two consecutive repeat cycles

    N is the number of epeat cycles

$R_{\oplus}$ is the mean equatorial Earth radius

- The increments to be done to the initial osculating eccentricity vector (involving eccentricity e and argument of perigee ω in order to obtain a more frozen orbit are derived as indicated in the frozen eccentricity algorithm in equations Eq. 7.7 and Eq. 7.8.
- Finally, the right ascension of the ascending node Ω and the true anomaly ω can be adjusted in order to ensure that the proposed Earth point is overflown at initial time, as described in previous section.

- Previous steps are iteratively repeated until mentioned differences vanish, and it is obtained a frozen reference orbit which implement a repeat pattern and keep constant the local time at node crossing.

7.2.7 Reference Orbit Synchronization After Spacecraft Injection

The reference orbit and consequently the reference ground track are referred to an initial time indicated by the user. This reference ground track plus a tolerance define the deadband in which the spacecraft should remain. If there is no constraint for the spacecraft to keep this initial phase indicated in the reference ground track (e.g. ERS-1), it is clear that this phase can be adjusted considering the time and orbital elements at the injection of the spacecraft.

Once that the injection state vector has been computed, it can be propagated backward up to the first ascending node crossing, and the corresponding longitude λ_0 will be selected. This will define the new origin of the reference orbit and the corresponding time, will be the one obtained in the backward propagation. Therefore the reference ground track has been synchronised considering the state of the spacecraft and time at injection and from now onwards the reference ground track and the spacecraft orbit shall have to maintain this synchronisation.

8 EARTH OBSERVATION ORBIT CONTROL

For Earth Observation Missions the target to be achieved is defined in one of the following ways:

- either the spacecraft should remain as long as possible inside of the deadband
- either the spacecraft should overflow certain Earth points at a prescribed time
- or a combination of both

Manoeuvres shall have to be planned and performed such that the spacecraft reaches the proposed target, from its current state vector. Taking the assumption that the orbit is near circular, the Gauss equations become as follows RD-8:

$$\begin{aligned}
&\Delta a \approx 2a\frac{\Delta V_T}{V} \\
&\Delta e_x \approx 2\frac{\Delta V_T}{V}\cos\alpha_0 - \frac{\Delta V_N}{V}\sin\alpha_0 \\
&\Delta e_y \approx 2\frac{\Delta V_T}{V}\sin\alpha_0 + \frac{\Delta V_N}{V}\cos\alpha_0 \\
&\Delta i \approx \frac{\Delta V_W}{V}\cos\alpha_0 \\
&\Delta\Omega \approx \frac{\Delta V_W}{V}\frac{\sin\alpha_0}{\sin i}
\end{aligned} \qquad \text{Eq. 8.1}$$

The following notations and assumptions are used in the equations:

- the velocity vector increments $\Delta\vec{V} = (\Delta V_T, \Delta V_N, \Delta V_W)$ qre represented by the three components, where (T, N, W) is the orthonormal reference frame which is defined as follows: its origin is the centre of mass of the satellite, the axis T is aligned in the direction of velocity, the axis W is aligned in the direction of the angular momentum vector of the orbit and the axis N completes the trihedral.
- the manoeuvres are performed with a single thrust, which is assumed to be instantaneous; the orbital point where the thrust occurs is located by its argument of latitude α_0.
- the semi-major axis a which is present in the first equation may, to a first approximation, be taken as constant and equal to the initial value a0 (mean parameter) before correction.

- the orbit is assumed to be near-circular (this is the case with frozen orbits) and therefore the second order terms in e in the original Gauss equations have thus been disregarded, and it is even possible to consider the velocity V as constant if the first order terms are disregarded.

$$V = V_R = \sqrt{\frac{\mu}{a}} \quad \Rightarrow \quad P = 2\pi \frac{a}{V_R} \qquad \text{Eq. 8.2}$$

- Δi and $\Delta\Omega$ are small.

These equations shall be used to compute the velocity vector increments.

8.1 Precession of the Orbital Plane

The precession of the orbital plane, for a near circular orbit is obtained as follows:

$$\dot{\Omega} = \frac{d\Omega}{dt} = -\frac{3}{2} J_2 \sqrt{\frac{\mu}{a^3}} \left(\frac{R}{a}\right)^3 \cos i \quad \Rightarrow \quad \dot{\Omega} = -FJ_2 \frac{V_R}{a^3} \cos i$$

where $F = \frac{3}{2} R^2$

The total precession for an orbit revolution is:

$$\Delta\Omega = \dot{\Omega}P = -\frac{FJ_2 2\pi \cos i}{a^2} \qquad \text{Eq. 8.3}$$

Resulting:

$$\frac{\partial}{\partial i}\Delta\Omega = \frac{2\pi FJ_2 \sin i}{a^2}$$
$$\frac{\partial}{\partial a}\Delta\Omega = \frac{4\pi FJ_2 \cos i}{a^3} \qquad \text{Eq. 8.4}$$

The change in the **precession**, after a time T, due to an **out-of-plane manoeuvre** parallel to the velocity vector ΔV_W at ascending (+) or descending (-) node can be evaluated as:

$$\left.\begin{aligned}\Delta i &= \pm\frac{\Delta V_W}{V_R}\\ \Delta\Omega &= \left(\frac{\partial}{\partial i}\Delta\Omega\right)N\Delta i\end{aligned}\right\} \Rightarrow \quad \Delta\Omega_{op} = \pm F J_2 \frac{\sin i}{a^3} T\Delta V_W \qquad \text{Eq. 8.5}$$

where N is the number of revolution in time T: $N = \frac{T}{P} = \frac{TV_R}{2\pi a}$

The change in the **precession**, after a time T, due to an **in-plane manoeuvre** parallel to the velocity vector ΔV_T can be evaluated as:

$$\left.\begin{aligned}\Delta a &= 2a\frac{\Delta V_T}{V_R}\\ \Delta\Omega &= \left(\frac{\partial}{\partial i}\Delta\Omega\right)N\Delta a\end{aligned}\right\} \Rightarrow \quad \Delta\Omega_{ip} = 4 F J_2 \frac{\cos i}{a^3} T\Delta V_T \qquad \text{Eq. 8.6}$$

8.2 Change in Longitude and Latitude

The change in **longitude** due to an **in-plane manoeuvre** ΔV_T can be evaluated as follows:

$$\Delta\lambda = \Delta\Omega_{ip} - \omega_E \Delta P \frac{T}{P}$$

where the first term of this equation $\Delta\Omega_{ip}$ is derived from the change in the precession due to an in-plane manoeuvre, and the second term is derived from the combination of the Earth's rotation and the change in the orbital period corresponding to the in-plane manoeuvre.

Considering that:

$$\frac{\Delta P}{P} \approx \frac{3}{2}\frac{\Delta a}{a} = 3\frac{\Delta V_T}{V_R}$$

The change in longitude can be finally evaluated as follows:

$$\Delta\lambda_{ip} = \left(4FJ_2\frac{\cos i}{a^3} - 3\frac{\omega_E}{V_R}\right)T\Delta V_T \qquad \text{Eq. 8.7}$$

The change in **longitude** due to an **out-of-plane manoeuvre** ΔV_W can be evaluated as follows:

$$\Delta\lambda = \Delta\Omega_{ip} = \pm F J_2 \frac{\sin i}{a^3} T \Delta V_W \qquad \text{Eq. 8.8}$$

In this case, because of the out-of-plane manoeuvre does not change the orbital period, the change in longitude is only affected by the change in the precession of the orbital plane.

The change in **latitude** due to an **in-plane manoeuvre** ΔV_T can be evaluated taking into account the relation between latitude δ and argument of latitude α=ω+ν, reflected in the following equation:

$$\sin\delta = \sin\alpha \sin i \qquad \text{Eq. 8.9}$$

Therefore, for a given latitude δ, it can be obtained the corresponding argument of latitude α as:

$$\alpha = \pm a\sin\left(\frac{\sin\delta}{\sin i}\right)$$

where the symbol (+) indicates the ascending part of the orbit and (-) the descending one.

The argument is directly related with the orbital period as:

$$\alpha = \alpha_0 + 2\pi N \quad \Rightarrow \quad \Delta\alpha = 2\pi\Delta N$$

And the change in the number of revolutions N due to an in-plane manoeuvre can be evaluated as:

$$N = \frac{T}{P} \quad \Rightarrow \quad \Delta N = -\frac{T}{P}\frac{\Delta P}{P} = -3\frac{T}{P}\frac{\Delta V_T}{V_R} = -\frac{3}{2\pi}\frac{1}{a} T \Delta V_T$$

and therefore the change in argument of latitude is obtained as:

$$\Delta\alpha_{ip} = -\frac{3}{a} T \Delta V_T$$

And the change in latitude now can be obtained using Eq. 8.9 as:

$$\Delta\delta = \frac{\cos\alpha}{\cos\delta} \sin i \Delta\alpha_{ip} \quad \Rightarrow \quad \Delta\delta_{ip} = -\frac{3}{a}\frac{\cos\alpha}{\cos\delta} T \Delta V_T$$

In first approximation, there is no change in latitude due to an out-of-plane manoeuvre ΔV_W. Because of the orbital period is not affected.

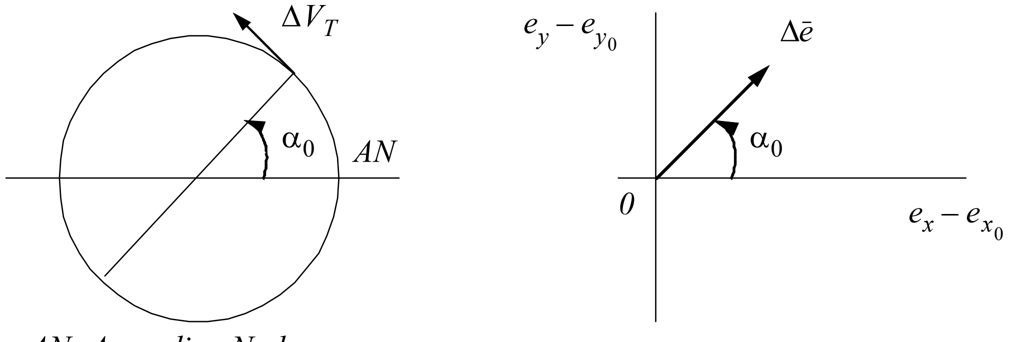
8.3 Frozen Eccentricity Algorithm

According to the Gauss equations Eq. 8.1, the efficiency of a thrust along the radial component ΔV_N, performed to correct the eccentricity vector, is half that of a correction made at a tangent to the velocity ΔV_T. Therefore it will be assumed that only the latter manoeuvres shall be considered, RD-8. In the following it will be referred in-plane manoeuvres to manoeuvres performed in the velocity direction ΔV_T. The Gauss equations become:

$$\Delta e_x = 2\frac{\Delta V_T}{V}\cos\alpha_0$$
$$\Delta e_y = 2\frac{\Delta V_T}{V}\sin\alpha_0 \qquad \text{Eq. 8.10}$$

Previous equations show that the change in the eccentricity vector due to an in-plane manoeuvre ΔV_T is a vector which direction is defined by the argument of latitude of the spacecraft α_0 at the manoeuvre time, the module is $2\frac{\Delta V_T}{V}$ and the sense depends on whether the ΔV_T is positive (in the sense of the spacecraft velocity vector) or negative (just opposite to the spacecraft velocity vector). The graphic interpretation is displayed below.

As derived from previous discussion, in order to obtain a predefined increment of the eccentricity vector $\Delta\vec{e}$, manoeuvres can be located at different argument of latitude α_0, but such that the sum of the change of eccentricity vector caused by each manoeuvre shall be the global eccentricity vector change $\Delta\vec{e}$.



AN: Ascending Node

It is clear that the optimum solution requiring less global ΔV, for the same eccentricity vector change $\Delta\vec{e} = (\Delta e_x, \Delta e_y)$, shall be the one with all in-plane manoeuvres ΔV_T performed at argument of latitude α or α+π, where α denotes:

$$\alpha = \operatorname{atan}\left(\frac{\Delta e_y}{\Delta e_x}\right)$$

In the following it will be assumed that all in-plane manoeuvres ΔV_T are performed at argument of latitude α or α+π.

The increments of eccentricity vector $\Delta\vec{e}$ shall be computed according to the theory described in RD-14, and briefly mentioned in a previous section of this document. Since the algorithm to compute the $\Delta\vec{e}$ is iterative, it implies that the computation of manoeuvres shall have to be also an iterative procedure. The algorithm consists of the following steps:

- An initial guess should be provided for the argument of latitude α. This can be obtained as the corresponding argument of latitude for the estimated time for the first in-plane manoeuvre ΔV_{T_1}.
- The orbit is propagated from the initial state up to the last manoeuvre. In the first iteration all in-plane manoeuvres are null.
- The mean eccentricity vector $\vec{e}_1$ of the first repeat cycle is obtained. This repeat cycle starts in the first descending node crossing after all manoeuvres are performed, and finishes in the closest descending node after a repeat cycle period. In the first iteration, it will be defined $\vec{e}_0 = \vec{e}_1$ representing the mean eccentricity vector without considering any manoeuvre.
- The mean eccentricity vector $\vec{e}_2$ of the second repeat cycle is obtained. This repeat cycle starts approximately at the end time of the first repeat cycle plus an integer number of repeat cycles, in the closest descending node, and finishes in the closest descending node after a repeat cycle period.
- For the first repeat cycle, $\Delta\vec{e}_0$ will be defined as the change in eccentricity vector due to the planned manoeuvres. Notice that at the firs iteration $\Delta\vec{e}_0 = \vec{0}$.

$$\Delta\vec{e}_0 = \left(\Delta e_{x_0}, \Delta e_{y_0}\right) = \vec{e}_1 - \vec{e}_0$$

$$\Delta e_0 = \left|\Delta\vec{e}_0\right|$$

$$\alpha_0 = \operatorname{atan}\left(\frac{\Delta e_{y_0}}{\Delta e_{x_0}}\right) - \alpha$$

If $\left|\alpha_0\right| > \frac{\pi}{2}$ then Δe_0 and α_0 shall be defined as:

$$\Delta e_0 = -\left|\Delta\vec{e}_0\right|$$

$\alpha_0 = \alpha_0 - \pi$

- The change in the eccentricity vector necessary to obtain a frozen orbit is calculated according to the theory described in RD-14. Firstly it is defined the increment in the mean eccentricity vector between the two considered repeat cycles:

  $$\Delta\vec{e}_1 = \left(\Delta e_{x_1}, \Delta e_{y_1}\right) = \vec{e}_2 - \vec{e}_1$$

  The change in the eccentricity vector $\Delta\vec{e} = \left(\Delta e_x, \Delta e_y\right)$ shall be:

  $$\Delta e_x = \frac{1}{2}\left(\Delta e_{x_1} - \Delta e_{y_1} \cot\left(\frac{At}{2|\vec{e}_1|}\right)\right)$$

  $$\Delta e_y = \frac{1}{2}\left(\Delta e_{y_1} + \Delta e_{x_1} \cot\left(\frac{At}{2|\vec{e}_1|}\right)\right)$$

  Adding this desired change in the eccentricity vector $\Delta\vec{e}$, to the change already obtained with the planned manoeuvres Δe_0, results in the mean eccentricity vector desired for the next iteration:

  $$\Delta\vec{e}_2 = \left(\Delta e_{x_2}, \Delta e_{y_2}\right) = \Delta\vec{e} + \Delta\vec{e}_0$$

  $$\Delta e_2 = |\Delta\vec{e}_2|$$

  $$\alpha_2 = \mathrm{atan}\left(\frac{\Delta e_{y_2}}{\Delta e_{x_2}}\right) - \alpha_0 - \alpha$$

  If $|\alpha_2| > \frac{\pi}{2}$ then Δe_2 and α_2 shall be defined as:

  $$\Delta e_2 = -|\Delta\vec{e}_2|$$

  $$\alpha_2 = \alpha_2 - \pi$$

  This α_2 is the increment to be done to the current argument of latitude for manoeuvres α. Therefore it is obvious that for next iteration the new value of the argument of latitude shall be $\alpha = \alpha_2 + \alpha$.

  The module of the change in the eccentricity vector shall be $\Delta e = \Delta e_2 - \Delta e_0$.

- This iterative process should be repeated until the change in the eccentricity vector Δe is small compared to Δe_0, and the change in argument of latitude α_2 is small compared with the argument of latitude α.

8.4 Inclination and Ascending Node Change

The Gauss equations Eq. 8.1 for inclination and right ascension of the ascending node change are:

$$\Delta i \approx \frac{\Delta V_W}{V} \cos\alpha_0$$

$$\Delta\Omega \approx \frac{\Delta V_W}{V} \frac{\sin\alpha_0}{\sin i}$$

From these equations it is clear that the most efficient manoeuvres to change the inclination must be performed at ascending or descending node $(\alpha_0 = 0; \alpha_0 = \pi)$, and to change the right ascension of the ascending node the most efficient manoeuvres have to be performed at $\left(\alpha_0 = \pm\frac{\pi}{2}\right)$.

For Earth Observation missions it is not expected to have big changes in Ω. Firstly because the launch window is restricted to have a good value of Ω at the injection, and secondly because the spacecraft is forced to remain in the deadband, and this one has been defined taking into account that the local time at node crossing should remain constant. Therefore it is only necessary small tuning of the Ω, to keep the local time at node crossing inside a tolerance. This can be obtained at the same time of the inclination change, since the precession of the orbital plane is affected by the inclination. For out-of-plane manoeuvres ΔV_W performed at ascending or descending node the change in Ω due to the precession of the orbital plane is given before in Eq. 8.5. Therefore it can be assumed that out-of-plane manoeuvres shall be performed to change the inclination, and therefore at ascending or descending node, but taking into account also the change in Ω.

For out-of-plane manoeuvres ΔV_W performed at node crossing, the change of inclination is obtained as:

$$\Delta i = \pm \frac{\Delta V_W}{V_R}$$

where (+) indicates that manoeuvre is at ascending node, and (-) at descending node.

The size of the out-of-plane manoeuvre ΔV_W to be performed, accounting for a desired inclination change, and for correcting the local time at node crossing can be stated as follows:

$$\Delta V_W = \Delta V_{W\Omega} \mp \Delta i V_R \qquad \text{Eq. 8.11}$$

where $\Delta V_{W\Omega}$ is the term to correct the local time at node crossing to be computed taking into account the precession of the orbital plane, and its calculation will be detailed in next section. The second term is to change properly the inclination.

Since the spacecraft is forced to remain in deadband, sometimes it is more useful to define the change in inclination, instead of using Δi, indicating a cross-track offset CTS to be obtained in the ground-track at orbital north pole. This is straight forward, since:

$$\Delta i = \frac{CTS}{R}$$

8.5 Manoeuvres Computation

The problem of the manoeuvres computation can be stated as to compute the timing and characteristics of a sequence of manoeuvres proposed by the user, in order to reach a defined target and fulfilling the problem constraints.

As mentioned in previous sections of this document some assumptions have been taken into account in order to simplify the problem. Hereafter the major ones are summarised:

- The orbit is assumed to be near circular.
- Manoeuvres are performed with a single thrust, which is assumed to be instantaneous (impulsive manoeuvres).
- The algorithm to compute the manoeuvres shall be iterative, since the algorithm to obtain a frozen orbit is iterative.
- Out-of-plane manoeuvres ΔV_W are performed at node crossing.
- Only in-plane manoeuvres, parallel to the spacecraft velocity vector, ΔV_T are to be used to the eccentricity vector control. In-plane manoeuvres along the radial component ΔV_N are not used since its efficiency is half that the first ones.
- In-plane manoeuvres ΔV_T are performed at an argument of latitude α or $\alpha+\pi$, being α the same for all manoeuvres.
- All manoeuvres are performed at the beginning of the period in study, in a reserved time interval. The mean eccentricity vectors required for the eccentricity algorithm are computed for a repeat cycles after the manoeuvres reserved period. Also the Earth's points to be overflown have to be referred to an epoch subsequent to this reserved period.

  The user shall propose a sequence of manoeuvres:
- A number $J \geq 0$ of out-of-plane manoeuvres ΔV_{W_j} with *k=1, ..., J*, indicating for each one the approximate time/node to be performed.

- A number $K \geq 0$ of in-plane manoeuvres ΔV_{T_k} with *j=1, ..., J*, indicating for each one the approximate time to be performed.

The location of the in-plane manoeuvres, as mentioned, is at argument of latitude α or α+π. Therefore, the number of unknown parameters is *K+J+1*.

The user shall also define certain targets to be achieved after this sequence of manoeuvres. The most important are:

- Desired change in inclination Δi, or a cross track offset *CTS* to be obtained at orbital north pole. It is clear that this change can always be achieved with a single out-of-plane manoeuvre ΔV_W given by Eq. 8.11.

However, due to manoeuvres constraints (e.g. maximum firing duration), sometimes it has to be broken down into several manoeuvres to be performed at consecutive ascending or descending nodes. In this case a criterion to split the manoeuvre should be considered. In fact, in case that *J>1*, the following should be satisfied:

$$\Delta V_W = \sum_{j=1}^{J} \Delta V_{W_j}$$

Therefore the number of unknown parameters becomes in *K+1*.

- Desired local time at node crossing. This will be used to compute the term $\Delta V_{W\Omega}$ introduced in previous section. The global change in the right ascension of the ascending node due to in-plane and out-of-plane manoeuvres is given by equations Eq. 8.5 and Eq. 8.6. The change in local time at node crossing is equivalent to a change in Ω. Therefore the following relation has to be fulfilled:

$$\sum_{k=1}^{K} 4FJ_2 \frac{\cos i}{a^3} T_K \Delta V_{T_K} \pm FJ_2 \frac{\sin i}{a^3} T \Delta V_{T\Omega} = \Delta\Omega = \Delta LT \qquad \text{Eq. 8.12}$$

where ΔLT is the change to be done in local time at ascending or descending node crossing.

- Desired *N* Earth's points to be overflown at an approximate time. Propagating up to this approximate time and afterwards up to the same latitude of the Earth point to be overflown, the shift in longitude $\Delta\lambda_n \left(n = 1,...,N\right)$ is obtained for each one. The necessary in- plane manoeuvres ΔV_{T_k}, accounting for equations Eq. 8.7 and Eq. 8.8, shall fulfil the following relation:

$$\sum_{k=1}^{K} \left(4FJ_2 \frac{\cos i}{a^3} - 3\frac{\omega_E}{V_R} \right) T_{K_n} \Delta V_{T_K} \pm FJ_2 \frac{\sin i}{a^3} T_n \Delta V_{W\Omega} = \Delta\lambda_n \qquad \text{Eq. 8.13}$$

Therefore the number of degrees of freedom becomes *K+1-N.*

- Desired *M* Earth's latitudes to be overflown at a prescribed time. It was described previously that these latitude can be easily converted to argument of latitude. Propagating until the prescribed time it can be obtained for each latitude the change to be done in argument of latitude $\Delta\alpha_m \left(m = 1,...,M\right)$. It is clear that the following relation has to be fulfilled:

$$-\sum_{k=1}^{K} \frac{3}{a} T_{K_m} \Delta V_{T_K} = \Delta\alpha_m \qquad \text{Eq. 8.14}$$

Therefore the number of degrees of freedom becomes *K+1-N-M.*

The in-plane manoeuvres ΔV_{T_K} are also constrained to obtain a frozen orbit. As discussed previously this implies two additional equations. The first one calculates the argument of latitude α in which all in plane manoeuvres should be performed. The second one establishes a relation between ΔV_{T_K} and the change required in the eccentricity vector Δe, as follows:

$$\sum_{k=1}^{K} \left(\pm \Delta V_{T_K}\right) = \frac{V_R}{2} \Delta e \qquad \text{Eq. 8.15}$$

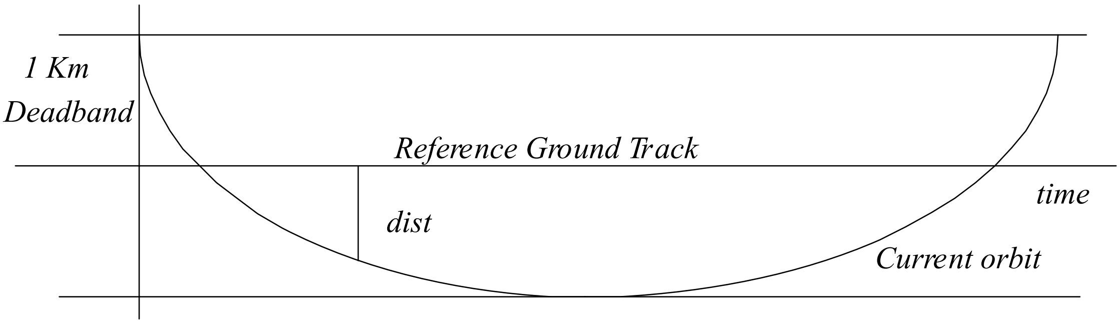
where the sign (+) corresponds to manoeuvres ΔV_{T_K} at argument of latitude α and the sign (-) to manoeuvres ΔV_{T_K} at argument of latitude α+π.

It is clear that the number of degrees of freedom become *K-N-M-1*. If this number is null, it means that the problem is completely defined. Therefore considering the system of simultaneous linear equations defined by equations Eq. 8.11, Eq. 8.12, Eq. 8.13, Eq. 8.14 and Eq. 8.15, the problem can be solved and the manoeuvres ΔV_W and ΔV_{T_K} can be obtained as well as the timing for each of them.

However if the number of degrees of freedom is greater than zero *(K-N-M-1>0)* the manoeuvres can also be computed taking into account an optimization criteria.

8.6 Deviation of S/C with respect to Reference Ground Track

The deviation of the S/C with respect the reference ground track, is defined as the perpendicular distance between the projection of the S/C position over the Earth's surface, and the reference ground track. In order to ease the computation it would be advisable to provide the reference ground track position in the Geodetic System and the velocity in Earth Fixed System.



Now the calculation of the deviation (*dist*) of the S/C with respect the reference ground track, can be formulated in two different manners:

- The first one would be to calculate such deviation for a given point of the ground track. Assuming that it is known the ground track velocity at that point, a plane can be defined passing through the ground track point and being perpendicular to the velocity (crossing plane). The orbit has to be propagated up to cross this plane, and the projection of the crossing point in the Earth's surface is derived. The deviation from the reference orbit will correspond to the distance between this projected point and the ground track.

  Let t_0 be the time to reach the ground track point defined as (*long*, *lat*) at the current cycle. The local vertical at that point is obtained as follows:

  $$\overline{GT_z} = (\cos lat \cdot \cos long, \cos lat \cdot \sin long, \sin lat)$$

  Let $\overline{GT_R}$ and $\overline{GT_V}$ be the position and velocity in Earth Fixed System of the ground track point. The plane, then will be defined by its normal unit vector:

  $$\overline{GT_n} = \frac{\overline{GT_V} - \left(\overline{GT_V} \cdot \overline{GT_z}\right)\overline{GT_z}}{\left|\overline{GT_V} - \left(\overline{GT_V} \cdot \overline{GT_z}\right)\overline{GT_z}\right|}$$

  Finally the right hand tried is completed by the unit vector, in which the distance will be measured:

  $$\overline{GT_t} = \frac{\overline{GT_n} \times \overline{GT_z}}{\left|\overline{GT_n} \times \overline{GT_z}\right|}$$

  In order to compute by when the orbit crosses the plane, an iterative process will be applied starting by $t = t_0$ and updating the time, and subsequently the orbit state, as:

  $$t = t - \frac{\overline{GT_n} \cdot \left(\overline{R_E} - \overline{GT_R}\right)}{\overline{V_E} \cdot \overline{GT_n}}$$

where $\overline{R_E}$, $\overline{V_E}$ are respectively the position and velocity of the spacecraft at time t in Earth Fixed System. Once the previous process has converged, it can be derived the projection of the corresponding orbit point over the Earth Surface $\overline{R_R}$. Now the distance between the orbit and the ground track is computed as:

$$dist = \overline{GT_t} \cdot \left(\overline{R_R} - \overline{GT_R} \right)$$

- The second one would be to calculate such deviation for a given S/C position. The problem now becomes in finding the ground track point, such that its crossing plane contains the S/C position. This can be done using an iterative process completely analogous to the one described in previous item. Once this ground track point has been found, the deviation from the ground track is computed as previously.

Notice that in both cases it is required that the S/C orbit and the reference ground track are synchronised.

8.7 Optimization of the S/C Remaining Time within Deadband

In previous discussion, the manoeuvres are determined by the fulfilment of the constraints. When there are some degrees of freedom, then is possible to consider optimisation criteria. In this case the problem can be formulated as “to compute the sequence of manoeuvres such that the spacecraft remains as long as possible within deadband”. This clearly defines the optimisation criteria. In order to complete the formulation of the problem, the constraints to be considered and assumptions to be taken into account have to be stated.

The optimisation problem will be solved using the OPXRQP optimisation routine (see [R18.]). For such, one has to define the function to be minimised, the equality and inequality constraints associated and if possible the partial derivatives of the function and constraints with respect the optimisation variables.

8.7.1 Manoeuvre Sequence

All assumptions mentioned in section 8.5 of this document are applicable to this optimisation problem. Manoeuvres are performed in a reserved period at the beginning of the optimisation. In-plane manoeuvres are performed at the same argument of latitude α or $\alpha+\pi$. The size of this in-plane manoeuvres will become in the optimisation variables.

8.7.2 Function to be Optimised

The optimisation criteria have been clearly stated previously. Now a function has to be defined supporting such criteria. In principle one can think that function to be optimised is precisely the remaining time in the deadband. This function has been demonstrated in the SR phase of NAPEOS (RD-17), not to be suitable for such optimisation, because it requires very long propagation periods and causes no convergence for some cases. The function to be used is the sum of the squares of the deviation of the orbit with respect the reference ground track. This distance will be evaluated at certain latitudes.

$$f = \sum_{i=1}^{I} \left(dist_i\right)^2 \qquad \text{Eq. 8.16}$$

It is clear that the minimisation of such a function, considering some constraints, is equivalent to maximising the remaining time within deadband. Such constraints will be mentioned in following section.

8.7.3 Constraints to be considered

The constraints concerning earth points or latitudes to be overflown and keeping frozen the eccentricity vector (Eq. 8.13, Eq. 8.14 and Eq. 8.15) are applicable to the optimisation. Constraints would be formulated as:

$$\begin{aligned} &\frac{V_R}{2}\Delta e = 0 \\ &\Delta\lambda_n = 0 \\ &\Delta\alpha_m = 0 \end{aligned} \qquad \text{Eq. 8.17}$$

The first constraint will be considered in case that more than one in-plane manoeuvre are to be calculated. The second and third set of constraints, only in case that the user is interested in overflying certain earth points or latitudes. This defines the equality constraints of the optimisation problem, which relation with the optimisation variables is expressed in equations (Eq. 8.13, Eq. 8.14 and Eq. 8.15).

Moreover some specific constraints must be taken also into account to consider additional characteristics such as:

- The spacecraft must remain within a deadband defined along the reference orbit. Therefore this can be managed as a set of constraints to be fulfilled. The minimum period in which the spacecraft should remain within a deadband can be defined. The deviation from the current orbit with respect the reference ground track can be calculated at points within the previously defined period. Constraints would be formulated as:

  $$\left(wd\right)^2 - \left(dist_k\right)^2 \geq 0 \qquad \text{Eq. 8.18}$$

  where *wd* is the deadband around the reference orbit.

- The westerly point of the current orbit, must be within the deadband, and moreover with a certain offset with respect the west limit of the deadband. This offset is computed in the calculation of the initial guess (initial value of the size of in-plane manoeuvres to start the Optimisation) that will be explained hereafter. A tolerance will be applied around this offset, and the offset of the westerly point of the spacecraft orbit will be forced to be within this tolerance.

$$\begin{aligned} & wd - offset + tol + dist_l \geq 0 \\ & -(wd - offset - tol) - dist_l \geq 0 \end{aligned} \qquad \text{Eq. 8.19}$$

These last equations will define the inequality constraints of the optimisation problem.

8.7.4 Partial Derivatives of the Function to be Optimised

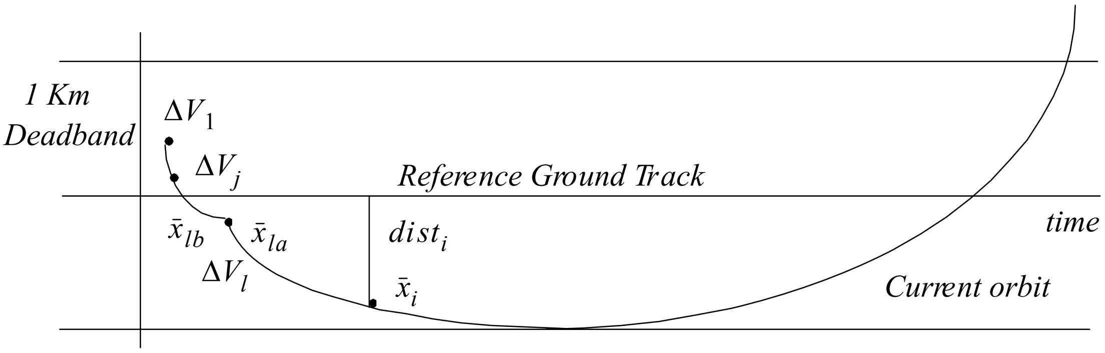
The function to be optimised is stated in equation Eq. 8.16.The partial derivatives of such function with respect the optimisation variables ΔV_j can be calculated as follows:

$$\frac{\partial(f)}{\partial(\Delta V_j)} = 2\sum_{i=1}^{I} dist_i \frac{\partial(dist_i)}{\partial(\Delta V_j)} \qquad \text{Eq. 8.20}$$

Now the partial derivative of the deviation with respect the optimisation variable can be calculated as follows:

$$\frac{\partial(dist_i)}{\partial\Delta V_j} = \frac{\partial(dist_i)}{\partial\vec{x}_i} \cdot \frac{\partial\vec{x}_i}{\partial\vec{x}_{la}} \cdot \frac{\partial\vec{x}_{la}}{\partial\vec{x}_{lb}} \cdot \frac{\partial\vec{x}_{lb}}{\partial\vec{x}_{ja}} \cdot \frac{\partial\vec{x}_{ja}}{\partial\Delta V_j} \qquad \text{Eq. 8.21}$$

In the graphic below is depicted the parameters involved in previous equation.



The calculation of this partial derivatives proceeds as follows:

- The partial derivative of the deviation $dist_i$ with respect to an state vector $\vec{x}_i$ is calculated as:

$$\frac{\partial(dist_i)}{\partial\vec{x}_i} = \begin{bmatrix} \frac{\partial(dist_i)}{\partial x_j} \\ \frac{\partial(dist_i)}{\partial y_j} \\ \frac{\partial(dist_i)}{\partial z_j} \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

  This previous partial derivatives:

  $$\frac{\partial(dist_i)}{\partial x_j}, \frac{\partial(dist_i)}{\partial y_j} \text{ and } \frac{\partial(dist_i)}{\partial z_j},$$

  can be evaluated by finite differences, applying an infinitesimal change in the state vector $\vec{x}_i$, and calculating the change in deviation $dist_i$.

- The partial derivative of one state vector with respect another state vector:

  $$\frac{\partial\vec{x}_i}{\partial\vec{x}_{la}}$$

  when there is no manoeuvre in between, is calculated in the propagator integrating the variational equations.

- The partial derivative of the state vector after a manoeuvre $\vec{x}_{la}$ with respect to the state before manoeuvre $\vec{x}_{lb}$ is the identity matrix, when considering impulsive manoeuvres.

- The partial derivative of the state vector after a manoeuvre $\vec{x}_{ja}$ with respect the size of the manoeuvre ΔV_j can be evaluated taking into account that for impulsive manoeuvres the following relation is held:

$$\vec{x}_{ja} = \vec{x}_{jb} + \Delta V_j \begin{bmatrix} \dfrac{0}{\Delta\overline{V}_j \big/ \left|\Delta\overline{V}_j\right|} \end{bmatrix}$$

Therefore:

$$\frac{\partial \vec{x}_{ja}}{\partial \Delta V_j} = \begin{bmatrix} 0 \\ \overline{\Delta V_j} \Big/ \left|\overline{\Delta V_j}\right| \end{bmatrix}$$

which is a vector with its first third components are zero, and the second third represents the manoeuvre direction, which for fixed attitude manoeuvres is a known vector.

8.7.5 Partials Derivatives of the Constraints

The equality constraints (Eq. 8.17) are related with the optimisation variables through the linear equations (Eq. 8.13, Eq. 8.14 and Eq. 8.15). Therefore the partial derivatives are calculated as follows:

$$\begin{aligned}
&\frac{\partial}{\partial \Delta V_j}\left(\frac{V_R}{2}\Delta e\right) = \pm 1 \\
&\frac{\partial}{\partial \Delta V_j}(\Delta\lambda_n) = \left(4FJ_2\frac{\cos i}{a^3} - 3\frac{\omega_E}{V_R}\right)T_{K_n} \\
&\frac{\partial}{\partial \Delta V_j}(\Delta\alpha_m) = -\frac{3}{a}T_{K_m}
\end{aligned} \tag{Eq. 8.22}$$

The inequality constraints (Eq. 8.18 and Eq. 8.19) are functions of the deviation $dist_i$. Their partial derivatives with respect to the optimisation variables are:

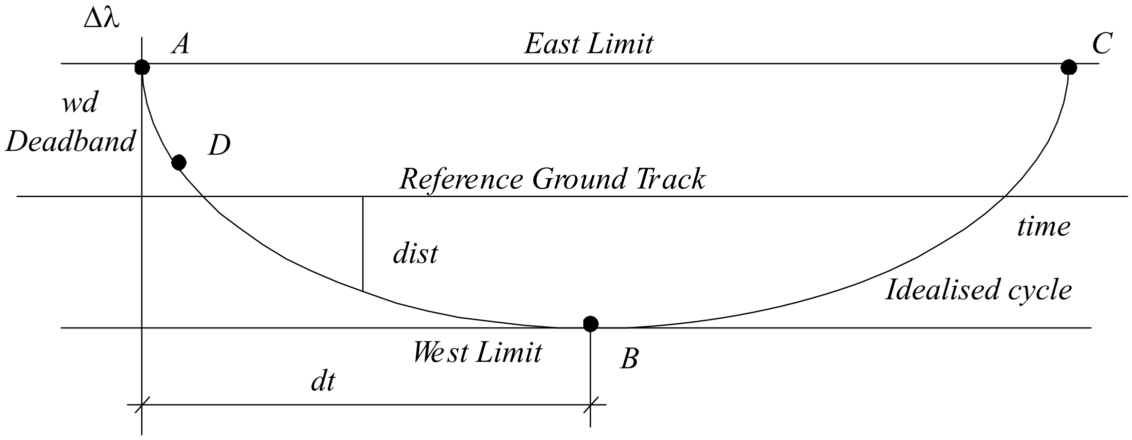
$$\begin{aligned}
&\frac{\partial}{\partial \Delta V_j}\left((wd)^2 - (dist_k)^2\right) = -2(dist_k)\frac{\partial}{\partial \Delta V_j}(dist_k) \\
&\frac{\partial}{\partial \Delta V_j}(wd - offset + tol + dist_l) = \frac{\partial}{\partial \Delta V_j}(dist_l) \\
&\frac{\partial}{\partial \Delta V_j}(-(wd - offset - tol) - dist_l) = \frac{\partial}{\partial \Delta V_j}(dist_l)
\end{aligned} \tag{Eq. 8.23}$$

And these partial derivatives of the deviation of the orbit with respect the size of the in-plane manoeuvres are calculated as indicated previously in equation Eq. 8.21.

8.7.6 Initial guess computation

In order to save iterations in the optimisation, it is advisable to provide an initial guess value of the optimisation variables. To compute these initial values, equations Eq. 8.12, Eq. 8.13, Eq. 8.14 and Eq. 8.15) are used. The problem has degrees of freedom and therefore, firstly some additional constraints must be included, just to solve the linear system. Obviously these new constraints will not be taken into account in the optimisation problem. These new constraints will be obtained considering additional Earth Points to be overflown, which are derived from the idealised behaviour of the spacecraft in the deadband.

The variation of the deviation of the spacecraft with respect the reference ground track, is mainly dominated by the air drag. In some cases when the air drag is small the impact of the inclination variations becomes more significant (see ref. [R19.]). An idealised cycle is defined such that the deviation of the spacecraft at equator with respect the reference ground track corresponds with the easterly deadband. Subsequently this deviation will vanish and increase up to the westerly deadband, to vanish again and increase up to the easterly deadband, as depicted in the following figure.



This can be obtained increasing the initial semi major axis above its nominal value (A), such that the air drag causes a Westward drift in ground track at equator. When the semi major axis reaches the nominal value, the spacecraft is in the West limit of the deadband (B). Subsequently, when the semi major axis decays below its nominal value the air drag causes a Eastward drift. When spacecraft reaches the East limit (C), it would be the time to perform a semi major axis raising manoeuvre.

This idealised cycle can be approximated by a parabola, since the change in longitude is calculated at first approximation by the equation:

$$\Delta\lambda = \left(4FJ_2 \frac{\cos i}{a^3} - 3\frac{\omega_E}{V_R}\right)\Delta t V_R \frac{\Delta a}{2a} \qquad \text{Eq. 8.24}$$

The change in semi-major axis can be approximated by:

$$\Delta a = \frac{da}{dt}\Delta t$$

Therefore, the change in longitude, that is equivalent to a change in the deviation with respect the reference ground track, is a quadratic function in time.

Based in Eq. 8.24 the parameter *dt* can be calculated. This parameter defines the idealised cycle for a prescribed deadband of width *2wd*.

$$\Delta\lambda = \frac{2wd}{R_E \sin i} \quad \Rightarrow \quad dt = \sqrt{\frac{4awd}{\left(4FJ_2 \frac{\cos i}{a^3} - 3\frac{\omega_E}{V_R}\right)V_R \frac{da}{dt} R_E \sin i}} \qquad \text{Eq. 8.25}$$

In previous equation the only unknown parameter is the variation of the semi major axis with respect time,

$$\frac{da}{dt}$$

which depends mainly on the air drag. To calculate accurately this parameter the orbit has to be propagated for a period of time. This period will coincide with the time necessary to calculate also the two average eccentricity vectors necessary for the frozen eccentricity vector constraint. The semi major axis a_i will be approximate using the least squares method:

$$\alpha\sum_i t_i^{\,2} + \beta\sum_i t_i = \sum_i a_i t_i$$

$$\alpha\sum_i t_i + \beta\sum_i 1 = \sum_i a_i$$

α and β are calculated solving previous linear equations system, and obviously:

$$\frac{da}{dt} = \alpha$$

The quadratic idealised cycle is calculated constraining the parabola to pass through points (A), (B) and (C), and becomes:

$$dist = \frac{2wd}{dt^2}t^2 - \frac{4wd}{dt}t + wd \qquad \text{Eq. 8.26}$$

Now the time of this idealised cycle (relative to the time of point (A)), must be corresponded with the current time of the optimisation problem. For such purpose the deviation (dist) of the orbit with respect the ground track after the sequence of manoeuvres t_{am} shall be calculated. By using equation Eq. 8.16 it is possible calculate the corresponding time for (idist) and then establish the relation between the time scale of the idealised cycle and the time of the optimisation problem.

Subsequently as many points of the ground track as degrees of freedom of the problem will be selected. The time for such points will be subsequent to t_{am}. The longitude and latitude of these points will be corrected by the predicted deviation at the corresponding time given by Eq. 8.16.

Now considering the new constraints (Earth points previously calculated to be overflown at calculated prescribed time), the linear system of equations (Eq. 8.22, Eq. 8.23, Eq. 8.24 and Eq. 8.25), can be solved and an initial value of the size of the in-plane manoeuvres can be derived.

Notice that this initial guess is as good as the calculated idealised cycle and equations Eq. 8.12, Eq. 8.13, Eq. 8.14 and Eq. 8.15 approximate respectively the real behaviour of the spacecraft in the deadband and the variations in longitude, eccentricity vector, etc, caused by in- plane manoeuvres. The first one is as good as the parameter:

$$\frac{da}{dt}$$

(mean value of the variation of semi major axis versus time) approximates the mean of the real evolution of this parameter versus time. It is clear that strong fluctuations of the semi major axis around the mean value will lead to a worse fitting of this idealised cycle to the real one, and therefore a degradation of the quality of the initial guess calculated.

9 RENDEZVOUS

9.1 Introduction

The rendezvous problem is the one defined as the calculation of timing and characteristics of the manoeuvres, to reach a certain state vector or orbital elements at a prescribed time, from the current spacecraft state. These manoeuvres shall be according to a manoeuvre model and be calculated considering an optimisation criteria and a set of constraints to be fulfilled. Therefore special attention must be paid to all of these items, in order to create an optimisation tool as generic as possible that can be applied to a different missions.

The optimisation problem shall be solved using the simplex method. For such purpose it is necessary to express the function to be minimised, as well as all equality and inequality constraints as a linear functions of the optimisation variables, and moreover it is obligatory that the optimisation variables are always greater equal than zero. The optimisation variables shall be the module of the delta-v to be applied. Moreover the mid-time of the manoeuvre is also an important parameter to be determined in the optimisation process.

Because of the cost function and constraints are linearised with respect a reference, the optimisation will consist of several iterations until convergence, to eliminate non-linear effects. For each iteration the reference around which to linearise the problem is the solution from previous one. Therefore it is clear that an initial guess is required to start the iterative process.

9.2 Manoeuvres Modelling

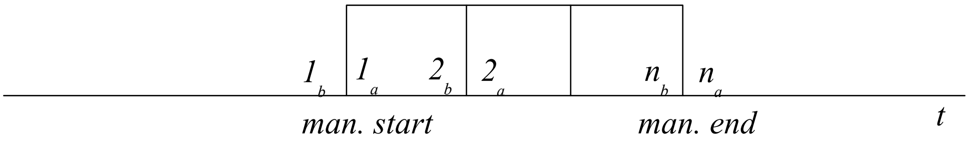
The rendezvous problem shall have to consider different types of the manoeuvres. The first classification would be impulsive or finite thrust manoeuvres. Both types are to be supported by the rendezvous. The first one have been already explained in section concerning Earth Observation, where they are used, therefore this section will explain the second type.

In absence of air drag force, the rocket equation becomes:

$$\Delta V = -\frac{1}{m}\int_{t_1}^{t_2} T dt - \int_{t_1}^{t_2} g \sin \gamma dt$$

where the first term of previous equation represents the increment in velocity due to a thrust profile provide by the thruster, and the second one represents the gravity losses. Therefore in order to obtain a certain ΔV the manoeuvre should last a time according to previous equation. This fact has to be taken into account in the orbital propagation, and also in the optimisation process. Now the partial derivatives of the state after the manoeuvre with respect to the ΔV

can be calculated assuming a series of impulsive manoeuvres during the time of the manoeuvre as depicted in following figure:



where the subscript b denotes before impulsive manoeuvre and a after manoeuvre. The global ΔV shall be equally distributed (can be also be done distributing the ΔV according to the thrust profile) between all impulsive manoeuvres. Therefore the partial derivative can be approximate by the following expression:

$$\frac{\partial \overline{x_{n_a}}}{\partial \Delta V} = \frac{\partial \overline{x_{n_a}}}{\partial \Delta V_n}\cdot\frac{\partial \Delta V_n}{\partial \Delta V} + \frac{\partial \overline{x_{n_a}}}{\partial \overline{x_{n_b}}}\cdot\frac{\partial \overline{x_{n_b}}}{\partial \overline{x_{n-1_a}}}\cdot\frac{\partial \overline{x_{n-1_a}}}{\partial \Delta V_{n-1}}\cdot\frac{\partial \Delta V_{n-1}}{\partial \Delta V}+\cdots+\frac{\partial \overline{x_{2_a}}}{\partial \overline{x_{2_b}}}\cdot\frac{\partial \overline{x_{2_b}}}{\partial \overline{x_{1_a}}}\cdot\frac{\partial \overline{x_{1_a}}}{\partial \Delta V_1}\cdot\frac{\partial \Delta V_1}{\partial \Delta V}$$

Therefore accounting for the well known relations for the impulsive case:

$$\frac{\partial \overline{x_{n_a}}}{\partial \Delta V_n} = \begin{bmatrix} 0 \\ I \end{bmatrix}$$

$$\frac{\partial \Delta V_n}{\partial \Delta V} = \frac{I}{n}$$

$$\frac{\partial \overline{x_{n_a}}}{\partial \overline{x_{n_b}}} = I$$

it becomes:

$$\frac{\partial \overline{x_{n_a}}}{\partial \Delta V} = \lim_{n\to\infty}\left\{\begin{bmatrix} 0 \\ \frac{I}{n} \end{bmatrix} + \left[\frac{\partial \overline{x_{n_b}}}{\partial \overline{x_{n-1_a}}}+\cdots+\frac{\partial \overline{x_{2_b}}}{\partial \overline{x_{1_a}}}\right]\begin{bmatrix} 0 \\ \frac{I}{n} \end{bmatrix}\right\}$$

Now the Now the manoeuvres can be performed in a well known attitude (fixed attitude case), or the attitude can be also be calculated as a result of the optimisation process (turn & burn case). The first type shall be considered in the rendezvous problem, and the second one are not described in the current formulation, although they can be included subsequently as an extension of the current problem, changing slightly its formulation.

For manoeuvres in which the spacecraft is changing the attitude concurrently the formulation of the problem would be more appropriate to be done in a system in which the manoeuvre is fixed (orbital system, body fixed, etc.), rather than the inertial system. In this case the transformation matrix to translate position and velocity from one to another are to be considered.

9.2.1 Optimisation Criteria

The optimisation criteria shall be to minimise the global fuel consumption or ΔV required for the manoeuvre sequence, respectively for finite thrust or impulsive manoeuvres.

Therefore the cost function can be formulated in terms of ΔV as follows:

$$C = \sum_{i=1}^{I} \Delta V_i$$

or formulated in terms of fuel consumption for finite thrust manoeuvres as:

$$C = \sum_{i=1}^{I} \frac{\Delta V_i}{V_{ex_i}}$$

where $V_{ex_i} = g_0 I_{sp_i}$ is the exhaust velocity at the time of the manoeuvre ΔV_i.

9.2.2 Constraints

The target to be achieved is expressed as a set of orbital elements or state vector to be achieved at prescribed time. This will be considered as equality constraints that shall be expressed as a linear function of the optimisation variables. The formulation of these constraints is as follows:

$$T = f(\Delta V) \quad \Rightarrow \quad T = f(\Delta V_0) + \frac{\partial f}{\partial \Delta V}(\Delta V - \Delta V_0)$$

and accounting for $T_0 = f(\Delta V_0)$, results in:

$$\frac{\partial f}{\partial \Delta V}\Delta V = T - T_0 + \frac{\partial f}{\partial \Delta V}\Delta V_0$$

where T is the target to be achieved, T_0 is the value of the target parameters at the target time in previous iteration, ΔV is the value in current iteration and ΔV_0 is the value in previous iteration.

The function f is obtained linearising around the reference as:

$$T = f(\Delta V) = T_0 + \sum_{i=1}^{I} \frac{\partial T}{\partial \Delta V_i} \Delta V_i$$

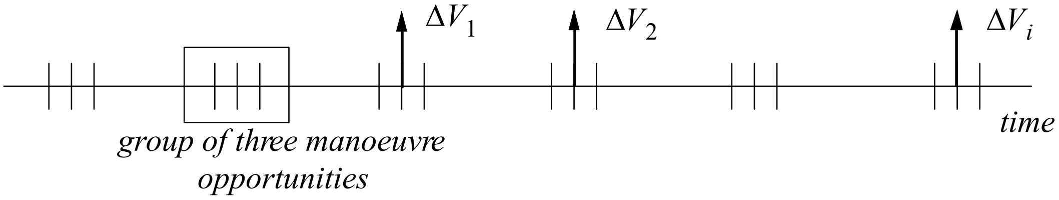
Moreover the rendezvous problem shall take into account a set of constraints, such as:

- Thruster performance, thruster configuration shall be taken into account in the manoeuvre modelling.

- Orbit manoeuvre location and operational constraints shall be taken into account in the time discretisation, as well as attitude modes of the spacecraft and platform constraints in the potential manoeuvres for each manoeuvre opportunity as explained in next section.

9.2.3 Time Discretisation. Manoeuvre Opportunities

For the simple method, the manoeuvres can only occur at discrete times. Each of one is called manoeuvre opportunity. They shall be grouped in a set of three consecutive manoeuvre opportunities. These groups will be defined at the first iteration such that will contain the initial manoeuvre sequence suggested. The rest of groups can be, for example, equally distributed in the gaps.



From iteration to iteration the number of these groups of manoeuvre opportunities will remain. Moreover the group of manoeuvre opportunities in which the size of the manoeuvres is zero remains at the same time, whilst the ones with at least one manoeuvre opportunity with size of ΔV_i different than zero will be shifted to be centred in the new manoeuvre time.

Now for each manoeuvre opportunity, all the allowed ΔV attitude are to be considered. Therefore considering n groups of manoeuvre opportunities and m allowed ΔV attitude the number of optimisation variables become to $3 \times n \times m$. It is clear that in order to compute accurate the time of the manoeuvre the discretisation should be as fine as possible, thus increasing the number of variables. Moreover if the number of allowed delta-v attitude is very big (i.e. turn & burn manoeuvres), the number of optimisation variables can not be managed efficiently by the simplex method. Therefore it is clear that other formulation of the problem is to be done in case of turn & burn manoeuvres.

9.2.4 Manoeuvres Calculation

To calculate the value of the ΔV_i, the simplex method shall be applied, well suited for the linear optimisation problem. This method is described in RD-15.

An important property of the solution of a linear optimisation problem is that the number of non-zero variables ΔV_i is equal or less than the number of active constraints. Therefore from iteration to iteration, from the $3 \times n \times m$ optimisation variables are different than zero at a maximum of K active constraints.

9.2.5 Manoeuvres Combination

As mentioned previously, at each iteration a number of ΔV_i different than zero are obtained. Mathematically it is possible to obtain these ΔV_i in the same manoeuvre opportunity group, and even at the same manoeuvre opportunity in case that several manoeuvre attitude are allowed. Clearly it does not have too much physical sense. In that case the manoeuvres with same attitude can be combined into a weighted manoeuvre, with size equal to the sum of sizes and the manoeuvre time a weighted balance between them. At next iteration the corresponding manoeuvre opportunity group shall be centred around this new time, as explained in previous section.

10 GNSS DATA PROCESSING

10.1 GNSS Observation Equations

Because GNSS data processing is one of the key capabilities within NAPEOS some basic information regarding the GNSS observations will be given in this chapter. We will, however, only review the most important aspects of the observation equations for GPS. For more information we refer to the open literature, e.g., Leick, RD-43.

Let us use the following notation:

k	Index of a particular receiver
i	Index of a particular transmitter
t	Signal reception time (GPS time)
t_k	Reading of the receiver clock at signal reception time
δ_k	Error of the receiver clock at time t with respect to GPS time
$t_k - \delta_k$	The signal reception time t
τ	Signal travelling time between satellite and receiver
δ^i	Error of the satellite clock at time $t - \tau$ w.r.t. GPS time
$\vec{r}_k(t)$	Position of receiver at signal reception time
$\vec{r}^{\,i}(t-\tau)$	Position of receiver at signal emission time $t - \tau$
$\delta^i{}_k$	Geometrical distance between satellite i at signal emission time $t - \tau$ and receiver k at signal reception time t.

GNSS systems provide different observations. Most commonly used are the code observations, also called pseudo ranges. However, for high accuracy applications we make use of the so-called carrier phase observations. These are based on measuring the phase of the carrier that is used to transport the codes from the satellite transmitter to the receiver antenna.

10.1.1 Code Observation Equation

Using the known codes provided on the GNSS carriers, the GNSS receivers measure the pseudo range observations. These are called pseudo ranges since on the receiver side the clock error is unknown and thus the measurement is not a pure range observation as in the case for Satellite Laser Ranging (SLR) observations. The observation equation may be written as:

$$P^i_{Fk}(t) = c((t+\delta_k) - (t-\tau) + \delta^i))$$

Using the geometrical distance δ^i_k the code observation equation, in its simplest form may be written as:

$$P^i_{Fk}(t) = \rho^i_k + c\delta_k - c\delta^i \qquad 10.1$$

P^i_{Fk} Is the code measurement at frequency F (in meters)

Note that the terms $c\delta_k, c\delta^i$ both depend on the frequency F due to the different locations of the phase centres of both transmitter and receiver antennae. Also interfrequency biases may be present.

10.1.2 Carrier Phase Observation Equation

The GNSS receiver measures the difference between two phases. The one phase transmitted by the GNSS satellites the other generated in the receiver. The observation equation may be written as:

$$\psi^i_{Fk}(t) = \phi_{Fk}(t) - \phi^i_F(t-\tau) + n^i_{Fk}$$

$\psi^i_{Fk}(t)$ Phase measurement at epoch t and frequency F (in cycles)

$\phi_{Fk}(t)$ Phase of the signal generated by the receiver oscillator at reception time t

$\phi^i_F(t-\tau)$ Phase of the signal generated by the satellite oscillator at emission time $t - \tau$

n^i_{Fk} Unknown integer number of cycles, the so-called initial phase ambiguity

Using a Taylor series development truncated after the terms of first order we may rewrite this:

$$\psi^i_{Fk}(t) = \phi_{Fk}(t) - \phi^i_F(t) + \tau \cdot f_F + n^i_{Fk}$$

Here f_F is the frequency of the carrier signal. Taking into account the receiver clock error δ_k and the satellite clock error δ^i the phase difference may be written as:

$$\phi_{Fk}(t) - \phi^i_F(t) = (\delta_k - \delta^i) f_F$$

The observation equation is then given by:

$$\psi^i_{Fk}(t) = (\delta_k - \delta^i) f_F + \tau \cdot f_F + n^i_{Fk}$$

Multiplying this equation by the wavelength λ_F we obtain the phase observation equation:

$$L^i_{Fk}(t) = \rho^i_k + c\delta_k - c\delta^i + \lambda_F n^i_{Fk} \qquad 10.2$$

10.1.3 Measurement Delays

Phase measurements and code pseudo-ranges are affected by both, systematic and random effects. Let us mention, e.g., satellite orbit errors, clock errors, propagation effects, receiver clock errors, relativistic effects, antenna phase centre variations, and multipath. Let us address three effects, namely tropospheric and ionospheric refraction, and the relativistic effect.

T^{i}_{k} Tropospheric refraction, i.e., the signal delay due to the neutral (i.e., the non-ionized) part of the Earth's atmosphere. It is important that the tropospheric refraction, in the GHz region, does not depend on the frequency and that the effect is the same for code and phase measurements (units: meters)

T^{i}_{k} Ionospheric refraction, i.e., the signal delay respectively phase advance due to the free electrons in the Earth's atmosphere. The ionosphere is a dispersive medium for microwave signals, which means that the refractive index for GPS signals is frequency-dependent (units: meters)

δ^{i}_{k} Relativistic correction. The atomic frequency standards in the GNSS satellites are "affected" by general (gravity) and special relativity (the satellite's velocity). The predominant portion of these relativistic effects is constant and, due to their common height, common to all GNSS satellites. The effect is that the clocks in orbit appear to run faster, for instance for GPS this would be about 40 seconds per day. This constant part is corrected for by adjusting the frequency of the oscillators of the GPS satellites prior to launch by -0.00455 Hz. The remaining part of the effect, caused by the eccentricity of the orbit, is often called the periodic relativistic effect (unit: seconds)

The periodic relativistic effect is not commonly used in the GNSS observation equations because it is satellite-dependent and therefore cancels in the single difference between stations. However, it is important if stations and/or satellite clocks are estimated, e.g., when applying undifferenced processing techniques and for absolute navigation. The effect, which may reach up to about 50 ns (15 m), is easily computed by the formula:

$$\delta_r^i(t) = F\cdot\sqrt{a(t)}\cdot e(t)\cdot\sin(E(t)) = \frac{-2\cdot\vec{r}^{\,i}(t)\cdot\dot{\vec{r}}^{\,i}(t)}{c^2}$$

Where F is a fundamental constant given by:

$$F = \frac{-2\cdot\sqrt{GM}}{c^2} = -4.4428\cdot10^{-10}\ {}^{s}\!/\!_{\sqrt{m}}$$

Where:

GM	Gravitational constant of the Earth
c	Velocity of light
a	Instantaneous semi-major axis of the satellite orbit

e	Eccentricity of the satellite orbit
E	Eccentric anomaly of the satellite
$\vec{r}^{\,i}(t), \dot{\vec{r}}^{\,i}(t)$	Geocentric position and velocity vector of the satellite *i*, respectively.

Taking into account these three effects the observation equations 10.1 and 10.2 for two different frequencies and both measurements (code and phase) may be refined as:

$$\begin{aligned} L_{1k}^{i} &= \rho_k^i - I_k^i + T_k^i - c\delta_r^i + c\delta_k - c\delta^i + \lambda_1 n_{1k}^i \\ L_{2k}^{i} &= \rho_k^i - \frac{f_1^2}{f_2^2} I_k^i + T_k^i - c\delta_r^i + c\delta_k - c\delta^i + \lambda_2 n_{21k}^i \\ P_{1k}^{i} &= \rho_k^i - I_k^i + T_k^i - c\delta_r^i + c\delta_k - c\delta^i \\ P_{2k}^{i} &= \rho_k^i - \frac{f_1^2}{f_2^2} I_k^i + T_k^i - c\delta_r^i + c\delta_k - c\delta^i \end{aligned} \qquad 10.3$$

10.2 GNSS Observation Differences

In GNSS data analysis it has become very common to form linear combinations and/or to generate difference between observations to reduce or eliminate some of the measurement delays or to reduce the number of unknown parameters.

Differences between the observations are usually formed to eliminate parameters, especially the receiver and satellite clocks and also the unknown phase ambiguities. For this purpose one can form single-, double-, and triple-differences. Differences that can be made are:

- **Between stations.** The difference between two (almost) simultaneous observations by two stations observing the same GNSS satellite. In this difference the satellite specific terms are (almost) perfectly eliminated, in particular the satellite clock and the relativistic delay ($\delta^i{}_{r\ and}\ \delta^i$). But also satellite specific biases are eliminated.
- **Between satellites.** The difference between two (almost) simultaneous observations by one station observing two different GNSS satellites. In this difference the station specific terms are (almost) perfectly eliminated, in particular the station clock (δ_k). But also other station specific biases are eliminated.
- **Between epochs.** The difference between two observations from one station observing one GNSS satellites on two different epochs. In this difference the initial phase ambiguity is eliminated. But also the constant receiver-transmitter biases are eliminated.

By using one of these differences a single differenced observation is formed, thus 3 different single differences are possible. By using two of these differences, two different ones of course, a double differenced observation is formed. There are also 3 different double difference observations possible. By using all three differences a triple differenced observation is formed. Only one triple difference combination may be formed.

If no differences are formed of the GNSS observations one typically calls this undifferenced or zero-difference processing.

Each difference, including the undifferenced one, has its pro's and con's. In NAPEOS we almost always used the undifferenced approach although the single and double difference processing should be possible.

10.3 GNSS Linear Combinations

GNSS signals are transmitted on different frequencies. This allows making different linear combinations of the independent signals from the different carriers. Currently only two frequencies are used by all GNSS systems but in the near future three frequencies will be used. The use of multiple frequencies was driven by the fact that the ionospheric delays are frequency dependent. With observations on two different frequencies the delay caused by the ionospheric refraction may be computed (to the first order) and removed from the observations. However, some more interesting linear combinations exist which we will briefly describe here.

With the advent of three frequencies and different observations on each of the frequencies much more linear combinations will become feasible. Some are very interesting as they, theoretically, should offer the almost instantaneous integer ambiguity resolution by using a technique called "triple frequency integer ambiguity resolution". However, this is currently beyond the scope of this document.

Here we limit ourselves to the linear combinations currently used in NAPEOS:

- Ionosphere-free Linear Combination (LC)
- Melbourne-Wubbena Linear Combination (MW)
- Wide-lane Linear Combinations (WL)
- Geometry-free linear combination (Iono)
- Single frequency Linear Combination (Graphics)

Linear combinations can be made on any difference level of the observations. Also linear combinations can be made out of a combination of code and phase observations.

In its most general form a linear combination of two code and two phase observations on different frequencies, LP_{LC}, may be written as:

$$LP_{LC} = \alpha \cdot L_1 + \beta \cdot L_2 + \gamma \cdot P_1 + \delta \cdot P_2 \qquad 10.4$$

Similarly a linear combination of only two phase observations on different frequencies, L_{LC}, may be written as:

$$L_{LC} = \alpha \cdot L_1 + \beta \cdot L_2 \qquad 10.5$$

One thing one should bear in mind is that there is a “price to be paid” when forming these linear combinations. This price is the increase of the noise of the observations. For the phase only linear combination the noise of combined observation may be expressed as:

$$\sigma_{L_{LC}} = \sqrt{\alpha^2 + \beta^2} \cdot \sigma_0$$

Here we have assumed, for simplicity, that the original observations, L_1 and $L_{2,}$ had the same noise, σ_o.

10.3.1 Ionosphere-free linear combination

The ionosphere-free linear combination is the one most widely used in GNSS processing and consequently also in NAPEOS. The large advantage here is that the effect of the ionosphere is removed, at least to the first order. The second order effect is at the few millimetre level only and can (still) be neglected. Nevertheless, in recent years models have been developed to account for the second order ionosphere effects.

For the Ionosphere-free linear combination we use 10.5 with the following values for α and β:

$$\alpha = \frac{f_1^2}{f_1^2 - f_2^2} \qquad \beta = -\frac{f_s^2}{f_1^2 - f_2^2}$$

Leading to the following observation equations for code and phase respectively:

$$L_{LC} = \frac{1}{f_1^2 - f_2^2}(f_1^2 L_1 - f_2^2 \cdot L_2) \qquad P_{LC} = \frac{1}{f_1^2 - f_2^2}(f_1^2 P_1 - f_2^2 \cdot P_2)$$

It should be pointed out that this linear combination does not have true wavelength. However, when in ambiguity fixing using the widelane to derive the difference between the number of cycles on L1 and L2 in a second step the ionosphere-free linear combination is used to resolve the true number of cycles on L1 and L2. In this step the ionosphere-free linear combination is often referred to as “narrowlane” combination. The term “narrow” in this case comes from the fact that the (artificial) wavelength of this linear combination for GPS is only 109 mm.

10.3.2 Melbourne-Wubbena linear combination (MW)

The Melbourne-Wubbena linear combination is widely used for observation screening and for integer ambiguity resolution. It is a combination of code and phase measurements which results in a combined measurement with a relatively long wavelength (e.g. 860 mm for GPS) and a noise that is lower than that of the individual code measurements. In fact, the combination is such that all information is lost and only an ambiguity and noise remains! All geometry, clocks, and ionosphere effects are removed! So if good code measurements are available this linear combination makes screening the data for cycle slips and outliers very easy as that it easily allows the detection of small cycle slips and outliers, e.g. at the 1 to 2 cycle level (1 to 2 m level). Because of this functionality it is also perfectly suited to estimate the integer value of the widelane ambiguities in the integer ambiguity resolution procedures.

For the Melbourne-Wubbena linear combination we use 10.4 with the following values for α, β, γ, and δ:

$$\alpha = \frac{f_1}{f_1 - f_2} \qquad \beta = -\frac{f_2}{f_1 - f_2} \qquad \gamma = -\frac{f_1}{f_1 + f_2} \qquad \delta = -\frac{f_2}{f_1 + f_2}$$

Leading to the following observation equation:

$$LP_{LC} = \frac{1}{f_1 - f_2}(f_1 L_1 - f_2 L_2) - \frac{1}{f_1 + f_2}(f_1 P_1 + f_2 P_2)$$

10.3.3 Widelane linear combination (WL)

The widelane linear combination is interesting for ambiguity resolution because just like the Melbourne-Wubbena combination it has a very long wavelength of 860 mm. This is also the reason why the wideline and the Melbourne-Wubbena linear combinations cause for some confusion. However, contrary to Melbourne-Wubbena the widelane combination uses phase only and it still contains the full geometry information and thus also the clock, troposphere, and ionospheric effects. Thus its use for ambiguity resolution is mainly on short(er) baselines where the ionosphere, and to a lesser extend the geometry, effects cancel out to a certain extent in the differencing. The widelane combination of the code observations does not really serve any purpose.

For the widelane linear combination we use 10.5 with the following values for α and β:

$$\alpha = \frac{f_1}{f_1 - f_2} \qquad \beta = -\frac{f_2}{f_1 - f_2}$$

Leading to the following observation equations for phase:

$$L_{WL} = \frac{1}{f_1 - f_2}(f_1\ L_1 - f_2 \cdot L_2)$$

10.3.4 Geometry free linear combination (IONO)

The geometry free linear combination, as the name says, cancels out all the geometry information leaving only, or mainly, the ionosphere effects. Thus this observation is ideally suited, and most commonly, used for the estimation of the state of the ionosphere. It is also well suited for the cleaning of undifferenced data.

For the geometry free linear combination we use 10.5 with the following values for *α* and *β*:

$$\alpha = 1 \qquad \beta = -1$$

Leading to the following observation equations for phase:

$$L_{IONO} = L_1 - L_2$$

10.3.5 Single frequency combination (Graphic)

This linear combination is of particular interest for space applications where the use of dual frequency receivers is too costly in terms of consumption of spacecraft resources. In case of single frequency code and phase observations this combination allows to eliminate all the ionospheric effect, but at the cost of having an ambiguity and more noise then a pure carrier-phase combination. Nevertheless, this combination is very helpful for low flying satellites that with single frequency receivers that still are significantly effected by the ionosphere.

For the single frequency linear combination we use 10.4 with the following values for *α*, *β*, *γ*, and *δ*:

$$\alpha = 1 \qquad \gamma = -1 \qquad \beta = \delta = 0$$

Leading to the following observation equation, called “Graphic”:

$$LP_{Graphic} = \frac{L_1 + P_1}{2}$$

10.4 Ambiguity fixing for GNSS estimation

It has been proved, that fixing the ambiguity estimates to their real integer values can improve the accuracy of the estimation process involving GNSS measurements and parameters (see RD-33 and RD-34).

10.4.1 Basic Ambiguity Resolution Concept

Integer ambiguity resolution can only be performed on double differenced ambiguities. Only in the double differenced data the uncalibrated delays in both the transmitters and the receivers are cancelled out. Thus only the double difference ambiguities may be considered to be integers.

The basic principle of our integer ambiguity resolution scheme is the following:

1 Baseline level

1.1 Form all double difference Melbourne-Wubbena widelane ambiguities

1.2 Sort and fix the Melbourne-Wubbena widelane ambiguities according to their fixing probability

1.3 Form the corresponding narrowlane ambiguities

1.4 Sort the ambiguities according to their fixing probability

1.5 Select the independent set of fixable narrowlane ambiguities

2 Network level

2.1 Sort all fixable ambiguities from the baseline level according to their fixing probability

2.2 Select the independent set of fixable ambiguities

3 (Re)solve the normal equation system

4 Next iteration or finish.

10.4.2 Baseline Level

10.4.2.1 *Forming the Melbourne-Wubbena widelane double difference ambiguities*

The first step in our ambiguity resolution is to build the Melbourne-Wubbena double difference ambiguities. To save computation time compared to the network method of Blewitt, RD-32, we will from the double difference ambiguities per baseline and we will limit the baseline length to a certain maximum. The maximum baseline length should be user definable and typical values will be around 4000 to 6000 km.

For the formation of the double difference (DD) ambiguities it would be good to ensure that the double difference ambiguity parameters have at least the equivalent of 3 double difference observations. Only DD-ambiguities with more than 3 observations should be considered candidates for fixing. However, since in Napeos it may be difficult to have and/or gather the number of double difference observations per double difference ambiguity, we will base the selection of the ambiguities on the information contained in the normal matrix. The normal matrix gives, in principle, equivalent information.

In this first step **all** possible, including redundant ones, DD-ambiguities on the selected baseline will be formed. Possible DD ambiguities are those ambiguities that have a non-zero value in their correlation field in the normal matrix. So only if all four involved undifferenced ambiguities are correlated a DD-ambiguity may be formed from them. Going through all possible double difference combinations and selecting only the valid one will give us the double-difference operator (**d**$_{\mathbf{w}}$). This double difference operator forms the DD-ambiguities and their variances according to the following equations. This double difference operator will have rows on which there are four ones two with a positive sign and two with a negative sign.

$$\mathbf{b}_{\mathbf{w}}^{\mathbf{dd}} = \mathbf{d}_{\mathbf{w}} \mathbf{b}_{\mathbf{w}}^{\mathbf{u}}$$

$$\boldsymbol{\sigma}_{\mathbf{w}}^{\mathbf{dd}} = \sqrt{\mathbf{d}_{\mathbf{w}} \boldsymbol{\sigma}_{\mathbf{w}}^{\mathbf{u}} \mathbf{d}_{\mathbf{w}}^{\mathbf{T}}}$$

Where:

$\mathbf{b}_{\mathbf{w}}^{\mathbf{dd}}, \mathbf{b}_{\mathbf{w}}^{\mathbf{u}}$	are the double differenced and undifferenced ambiguity, respectively.
$\boldsymbol{\sigma}_{\mathbf{w}}^{\mathbf{dd}}, \boldsymbol{\sigma}_{\mathbf{w}}^{\mathbf{u}}$	are the double differenced and undifferenced variance, respectively.

The undifferenced ambiguity and its variance follow from the Melbourne-Wubbena data. The double differenced ambiguities and their variances follow from the above formula.

10.4.2.2 *Sorting and fixing the Melbourne-Wubbena widelane ambiguities*

The Melbourne-Wubbena widelane DD-ambiguities should now be fixed depending on their fixing probability. For this purpose we will us the probability function P_0:

$$P_0 = 1 - \sum_{n=1}^{\infty} erfc\left(\frac{n-(b-I)}{\sqrt{2}\sigma}\right) - erfc\left(\frac{n+(b-I)}{\sqrt{2}\sigma}\right)$$

With:

$$erfc(x) = \frac{2}{\sqrt{\pi}} \int_x^{\infty} e^{-t^2} dt$$

b Widelane DD-ambiguity estimate in widelane cycles

I Nearest integer of widelane DD-ambiguity estimate

σ Widelane DD-ambiguity sigma in widelane cycles

All those widelane DD-ambiguities with a fixing probability larger then a user defined value will be resolved. Typical values would be between 0.99 and 0.9999. For safety purposes it will be necessary to define a minimal value of σ or a maximum value of $b-I$. A good solution for this is to take $\sigma = \left|\frac{b-I}{2}\right|$ if the initial value of σ is smaller than this value.

Note: On the ESOC Sun platform the "erfc" function is available in the Fortran libraries. Furthermore, for the summation over "n" it should be enough to go to n=6.

10.4.2.3 *Forming the narrowlane double difference ambiguities*

As the next step we now have to generate the narrowlane double difference ambiguities. Of course only those narrowlane DD-ambiguities for which the widelane ambiguity resolution was successful should be considered in this step. So the double-difference operator derived for the widelane DD-ambiguities should be updated so that only the fixable DD-ambiguities are generated. This will give us the double-difference operator for forming the ionosphere free DD-ambiguities (**d**_c) and their variances:

$$\mathbf{b}_{\mathbf{c}}^{\mathbf{dd}} = \mathbf{d}_{\mathbf{c}}\mathbf{b}_{\mathbf{c}}^{\mathbf{u}}$$

$$\boldsymbol{\sigma}_{\mathbf{c}}^{\mathbf{dd}} = \sqrt{\mathbf{d}_{\mathbf{c}}\boldsymbol{\sigma}_{\mathbf{c}}^{\mathbf{u}}\mathbf{d}_{\mathbf{c}}^{\mathbf{T}}}$$

Where

$\mathbf{b}_{\mathbf{c}}^{\mathbf{dd}}, \mathbf{b}_{\mathbf{c}}^{\mathbf{u}}$ are the double differenced and undifferenced ambiguity, respectively.

$\boldsymbol{\sigma}_{\mathbf{c}}^{\mathbf{dd}}, \boldsymbol{\sigma}_{\mathbf{c}}^{\mathbf{u}}$ are the double differenced and undifferenced variance, respectively.

The subscript (**c**) is used to identify that these ambiguities are based on the ionosphere free linear combination.

The narrowlane DD-ambiguity and its variance may now be derived from the widelane and the ionosphere free ambiguities. They can be written as:

$$b_n^{dd} = \frac{f_1+f_2}{f_1} b_c^{dd} - \frac{f_2}{f_1-f_2} b_w^{dd}$$

$$\sigma_n^{dd} = \frac{f_1+f_2}{f_1} \sigma_c^{dd}$$

Where f_1 and f_2 are the frequencies of the L1 and L2 carrier-phases, respectively.

Now the fixing probability is computed for all narrowlane DD-ambiguities. For this purpose we will use the same probability function as for the widelane ambiguities. Only those narrowlane DD-ambiguities that satisfy the, user defined, probability requirement are resolved. This gives us $\mathbf{b}_{\mathbf{n}}^{\mathbf{dd}}, \mathbf{b}_{\mathbf{c}}^{\mathbf{dd}}, \mathbf{d}_{\mathbf{c}}$

10.4.2.4 *Sorting the fixed ambiguities*

We now have a set of ambiguities for which both the widelane and the narrowlane ambiguity were resolved. Consequently we have to sort these fixed DD-ambiguities depending on their fixing probability. For this sorting we could use just the narrowlane fixing probability or the combined fixing probability based on the product of the widelane and narrowlane fixing probabilities. For the initial design the product of the widelane and narrowlane fixing probability will be used.

10.4.2.5 *Select the independent ambiguities*

Because we formed all possible DD-ambiguities we may, and will, have several redundant ambiguities in our sorted list of fixed DD-ambiguities. We therefore now have to select the independent subset of fixed DD-ambiguities from the sorted list. We start with the DD-ambiguity with the largest fixing probability. The next DD-ambiguity is checked for independency. If it is independent it will be added to the list of fixed DD-ambiguities otherwise it will have to be rejected. In this way we have to cycle through all selected DD-ambiguities.

For the independency check the modified Gram-Schmidt method is used. The standard algorithm has bad numerical properties, which show in this problem due to the dimensions of the vectors to check.

10.4.3 Network Level

10.4.3.1 *Standard algorithm*

Sorting the ambiguities

After going through all the possible baselines in the previously described manner we will have a large set of fixable DD-ambiguities. Because we have made all possible baselines we again may, and will, have some redundant ambiguities in the set of fixable ambiguities. These ambiguities are now again sorted based on their fixing probability. Note that for this purpose we reuse the same fixing probability value that was determined on the baseline level!

Select the independent set

Similar as on the baseline level we again start with the DD-ambiguity with the largest fixing probability. The next DD-ambiguity is checked for independency. If it is independent it will be added to the list of fixed DD-ambiguities otherwise it will have to be rejected. In this way we cycle through the complete list of ambiguities.

For the independency check the Gram-Schmidt method is used.

The selected double differences will give us the double difference operator ($\mathbf{d_c}$). However, the number of selected double differences (m) will always be lower than the number of undifferenced ambiguities (n). For later usage our double difference operator matrix must be an invertible matrix (size n x n). To achieve this $\mathbf{d_c}$ is filled with those undifferenced ambiguities which satisfy the independency check. Thus we add ones on the diagonal of $\mathbf{d_c}$ until we can not find any more independent undifferenced ambiguities. This will give us an invertible matrix $\mathbf{d_c}$ of size n x n.

10.4.3.2 *Simplified algorithm*

The standard algorithm requires a lot of CPU time to complete. The main bottleneck is the check of the linear independence. These are basically operations on a matrix of dimension the number of ambiguities, which for a normal set-up is more than 3000. To improve this aspect, a simplified algorithm was developed. It basically reduces the dimension of the vector set, by extending the check on the baseline level. Once all the baselines are computed independently of each other, they are sorted depending on the number of fixed ambiguities (first in the list the one with more fixed ambiguities). To ensure independency among baselines, they are picked from this sorted list and the ones independent with respect to the ground stations are chosen. This check is performed by keeping track of the stations used by the baselines as the sorted list is followed. A baseline is considered independent if any of its stations is not yet used, on the other side if both stations have been already used, the baseline is not independent. The resulting list of independent baselines (the one used after the first iteration) will be used for the iterative independence check.

Iterative independence check

Per baseline there will be, except in the first baseline, three types of independent vectors:

- The first one will contain the independent and the fixed ambiguities of previous baselines that are used by the current baseline. These ambiguities will be named reference ambiguities. They will be added to the base, and hence form the first independent vector of the baseline of the base. These elements will define the independence status of the DDs.
- The second one will contain the independent DDs of the current baselines (i.e. independent with respect to the DDs themselves, and to the reference ambiguities).
- The third one will contain the independent individual ambiguities used by the current baseline, taking into account the dimension of the baseline, the reference ambiguities

and the independent DDs. They are needed to generate a full-rank base (i.e. its dimension is equal to the number of ambiguities used by the baseline).

For the first baseline, the first type does not exist. Applying the iterative independence check per baseline, taking into account the above described types of ambiguities, the space dimension of each check is reduced to the number of ambiguities used by the baseline, speeding up the whole process dramatically. Using the concept of reference ambiguities, the independence of the DDs of different baselines is guaranteed when they are put together and used for constraining when solving the normal equation system.

10.4.4 Solving the Normal Equation System

After going through both the baseline and network steps previously described we have determined the integer resolved value of the double difference ionosphere free ambiguities ($\mathbf{b}_c^{dd}$) and we also have determined the double difference operator (**d**$_c$) (not a square matrix, just the linear independent DDs). With this information we now go back to our normal equation system to compute our "ambiguity fixed" solution.

The equations describing the independent DDs (i.e. the equalities of the DDs to the delta needed to make them integer, already computed in previous steps, when checking the iterative independence check) are used as constraints to the normal equation system, so the integer DD value is ensured, and new values of parameters are obtained. So starting from the NEQ system and the constraints:

$$\begin{bmatrix} \mathbf{N}_{11} & \mathbf{N}_{12} \\ \mathbf{N}_{21} & \mathbf{N}_{22} \end{bmatrix} \begin{bmatrix} \mathbf{x}_1 \\ \mathbf{x}_2 \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \end{bmatrix}$$

$$DD\, x_2 = \Delta dd$$

Being x$_2$ the ambiguities used by the remaining DDs and Δdd the delta to be applied to the original DDs to make them integer in cycles. Now to constraint the system, the constraining equations will be added to the NEQ system as observations, hence the resulting system is:

$$\begin{bmatrix} \mathbf{N}_{11} & \mathbf{N}_{12} \\ \mathbf{N}_{21} & \mathbf{N}_{22} + \dfrac{DD^T DD}{\sigma^2} \end{bmatrix} \begin{bmatrix} \mathbf{x}_1 \\ \mathbf{x}_2 \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 + \dfrac{DD^T \Delta dd}{\sigma^2} \end{bmatrix}$$

Being σ a small value to give the observation/constraint a strong weighting so it is correctly taken into account. Care must be paid when choosing this value to avoid numerical problems when solving the system.

Right afterwards, the NEQ system (without constraints) is modified to have as new initial values for ambiguities the ones obtained after solving the constrained system. The new values are the input to start the next iteration.

10.4.5 Iteration in the Ambiguity Fixing

It has been shown that ambiguity resolution improves the results of GPS solutions by a factor of two compare to non resolved solutions. However, it is has also been shown that ambiguities are easier to solve over short baselines than over long baselines. This depends on the quality of the solution. So it therefore makes sense to start ambiguity fixing using mainly short baselines. In a next iteration the solution will have improved thanks to the fixed ambiguities on the short baselines. Thus the ambiguity fixing will be easier and more successful for the long baselines as well. We therefore envision doing a couple of iterations where we increase the maximum baseline length from iteration to iteration. However, it might also be possible to get similar results by just iterating since the sigma's, and therefore the fixing probability, of the remaining ambiguities will improve. However, in the case of the simplified algorithm, and to ensure consistency in the ambiguities fixed in following iterations this modification in the base length is not applied. The inconsistency could appear if a previously computed baseline is not taken into account in the following iteration, then it is possible that a fixed ambiguity cannot be fixed anymore. To avoid this possibility the baselines used in the algorithm are always the same and computed during the first iteration, so their length is fixed to the initial values.

The iteration scheme stops if any of the following cases occurs:

- No more ambiguities remaining to be resolved
- No additional ambiguities are resolved in the last iteration
- The user defined maximum number of iterations is reached

The parameter estimates are written to the normal output files, following the NEQ file format.

11 EARTH ORIENTATION PARAMETERS

11.1 Introduction

This section describes the handling of the Earth Orientation Parameters in Napeos. Note that in the following convention is used to distinguish between EOP and ERP:

- **EOP = Earth Orientation Parameters**: The full set of parameters needed to describe the orientation of the Earth in the celestial reference frame. So this includes X-pole, Y-pole, UT1-UTC, but also the two Nutation angles.
- **ERP = Earth Rotation Parameters.** The subset of three angles describing position of the Earth rotation axis on the Earth surface: X-pole, Y-pole, and UT1-UTC.

Of course three angles would be sufficient to describe the orientation of the Earth in the celestial reference frame. However, there are a couple of reasons for using 5 angles rather than two. The two main reasons being:

- Because the dynamics of the Earth are so well known, nutation (and precession) can be calculated within seconds of arc over periods of many decades. Whereas the other three angles can not be predicted very accurately. So there is no great need to estimate the nutation angles.
- Space geodetic observations are not sensitive to the two nutation angles because we can not separate them from a rotation of the satellite orbit. Nutation angles are rotations of the (quasi) Inertial Earth-fixed axis. Space geodetic observations are, however, sensitive to the nutation rates! In fact, the same holds for UT1-UTC where space geodetic observations are only sensitive to the rate of UT1-UTC and not UT1-UTC itself because of its 1:1 correlation with the ascending node of the satellite orbit(s) and a Z-rotation the whole reference frame as defined by the observing stations.

The EOPs are used to transform from the Celestial Reference System (CRS) reference frame to the Terrestrial Reference System (TRS) reference frame and vice-versa. The CRS we use in Napeos is the J2000 reference frame as defined in the IERS Conventions (2003), RD-7. The TRS we use is the International Terrestrial Reference System (ITRS) as also defined in the same IERS Conventions. To realise the ITRS one has to use the International Terrestrial Reference Frame (ITRF) solutions. At the moment the latest ITRF solution is the ITRF2005. A new ITRF should become available by the end of 2009 and will be called the ITRF2008 (the year reflects the date of the latest data that was used in the ITRF realisation).

The transformation from CRS to TRS is, as given in the IERS Conventions:

$$[\mathbf{CSR}] = \boldsymbol{Q}(t)\, \boldsymbol{R}(t)\, \boldsymbol{W}(t)\, [\mathbf{TRS}]$$

Where $\boldsymbol{Q}(t)$, $\boldsymbol{R}(t)$, and $\boldsymbol{W}(t)$ are the transformation matrices arising from the motion of the celestial pole in the celestial system, from the rotation of the Earth around the axis of the pole, and from polar motions respectively.

The CRS to TRS transformation in Napeos follows the IERS Conventions and uses the software as provided by the IERS. The IERS2003 conventions give three methods to implement the software (method 1, 2a, and 2b). From these methods Napeos has followed the method 1 which is in consistent with the IAU 2000A Precession-Nutation model. The main difference with the method 2 is that this method uses the new (X, Y, s, θ) transformation rather then the classical, equinox-based, transformation (dψ, dε). This is important to note because the IERS now provides two different pole files; one pole file that follows the new IAU2000A transformation and one that follows the classical transformation.

Another issue which is important to know is that the IERS software turned out to be very computational intensive, in particular the NU2000A subroutine. It was therefore needed to “buffer” the information that is required to perform the transformation. So when the EOP information is initialized in Napeos a buffer is filled which stores the required information. This makes things a bit more complicated, as we will see later, but it reduced the CPU time from 10 hours to 1 hour. So it was a very much needed change.

11.2 EOP handling in Napeos

11.2.1 Conversion of IERS EOP information

The IERS provides several different sets of EOP information. The basic content of the different products is very similar. For detailed information on the different IERS products please visit the IERS web site at: http://www.iers.org

All IERS EOP basically contain the same information: X-pole, Y-pole, UT1-UTC, and the celestial pole offset dψ and dε for the IAU1980 model, or dX, dY for the IAU2000 model.

The difference between the sets is the availability, prediction time span, and consistency.

In GNSS we typically use the so called “Rapid data and predictions” data also called the IERS “Bulletin A” and which is being generated at the United States Naval Observatory (USNO). These are the files that we download from USNO. There are two variants of these files one following the IAU1980 (“finals.daily” and “finals.data”) and the other the IAU2000 “finals2000A.daily” and “finals2000A.data”.

Another interesting set of products are the long term Earth orientation data. This data series, which is generated at the “Observatoire de Paris”, is referred to as the IERS C04 series. The latest release of this series is called the EOP 05 C04. The special value of this series is that it should be fully consistent with the ITRF2005. However, the differences between the Bulletin A and the C04 series are really small. Also from the C04 series there are two variants, one following the IAU1980 and the other the IAU2000.

In addition to these EOP series the IERS is also responsible for the leap seconds. These are disseminated using the so called Bulletin C. USNO keeps the leap seconds in a special file called “tai-utc.dat” which is easier to read by a computer.

For our GNSS work we routinely download the IERS Bulletin A files finals2000A.daily, finals2000a.data, and tai-utc.dat from USNO and convert them to the Napeos internal format using the Napeos program IersConv. The IersConv utility can also convert the IERS EOP 05 C04 files but this is not done routinely. The Napeos internal EOP file contains X-pole, Y-pole, UT1-TAI, dX, and dY.

11.2.2 NAPEOS EOP initialisation

When a NAPEOS program needs EOP information the information is initialized in the following way:

- **DBcb_Init:** General initialisation routine
- **DBcb_ReadEop:** This routine reads the Napeos internal EOP file and stores it in the “eop” array
- **ErpNutInit:** This routine buffers the ERP and the Nutation and Precession information. It calls the DBcb_InitEop in two loops. The first loop is to fill the ERP buffer. The second loop is to fill the Nutation buffer.
- **DBcb_ErpBuf:** This is the ERP buffer that contains: time, X-pole, Y-pole, UT1-TAI. The buffering interval is 1800 seconds (0.5 hours).
- **DBcb_NutBuf:** This is the nutation buffer that contains: time, the precession nutation matrix, the precssion matrix, the nutation matrix, and dX, and dY. The buffering interval is 3600 seconds (1 hour).

As mentioned before the EOP buffering was needed to save a significant amount of CPU time. The buffering interval was chosen after careful testing that the actual behaviour of the parameters is accurately reproduced by the interpolation. The most critical points are the “boundary” values, i.e., those epochs for which we have external (IERS) values. This is because when generating the buffer the available IERS values (typically one every day at midnight) are interpolated linearly. Then the different model contributions are added to these linearly interpolated values. The models are, e.g., the IAU2000A nutation model, the sub-daily polar motion, the UT1 to UT1r correction etc. etc. The resulting values are stored in the buffer and later interpolated quadratically. So at the epochs very close to the IERS epoch (typically midnight) some “side effects” of the linear versus quadratic interpolation are visible. However, they are well below the accuracy level and thus are unproblematic.

In any case a lot of time is saved by the buffering. Originally it was only done for the nutation and precession because of the computational load of the NU2000A routine. Later on it was also done for the ERP values because it turned out that the sub-daily ERP model and the UT1-UT1R corrections also use a significant amount of CPU time.

11.2.3 NAPEOS EOP USAGE

When a Napeos program needs the EOP information, e.g., to perform a transformation the values in the EEP and/or Nutation buffers are used. To get the values for the correct epoch the values in the buffers are interpolated quadratically using the 3 points closest to the time of the request.

The Nutation buffer interpolation is done in the function:

- InterpolNutBuf in module ATcb_conv
- This function is only used by the routines
  - ATcb_SetMatrix
  - ATcb_TleVec
- InterpolErpBuf in module ATcb_conv

The ERP buffer interpolation is done in the function:

- InterpolErpBuf in module DBcb_data
- This function is only used by the subroutine
  - DBcb_IntEop

The nutation information is only needed when making a transformation from CRS to TRS or vice versa. Therefore, it is only needed in the ATcb module which deals with these transformations. So it is used in the “ATcb_SetMatrix” subroutine for “normal” transformations.

The EOP information is needed in several places. So in order to access and interpolate the DBcb_ErpBuf values to the required epoch the subroutine DBcb_IntEop is available. For programmes where the EOP information was not buffered the subroutine DBcb_InitEop is available.

11.3 NAPEOS ERP Estimation in Bahn

11.3.1 Introduction

ERP parameters may be estimated in Bahn. The ERP estimation functionality is:

- Estimate several sets of ERP parameters per Bahn run. This is interesting for multi-day solutions like used for SLR and DORIS. For GNSS processing it allows to estimate ERPs with a higher time resolution.

- The reference time is the middle of the estimation interval of the ERP set rounded to the nearest UTC hour of the day. All three parameter times, start-, middle-, and end-time, are stored in the NEQ file.
- The a priori values are computed for the derived reference time. After the estimation the a priori ERP values, as stored in the **DBcb_eop** array in **DBcb_data** are updated using the estimates. If the propagation time interval is larger then the determination interval the estimated ERP values are extrapolated using the estimated offset and rates.
- We using **EopUpd** to generate an EOP file there is a small issue that the times used in the NAPEOS EOP files have an hourly resolution and are given in UTC. A typical GNSS run uses a time window from 0:00 hours in GPS time meaning a 15 second time offset in UTC time. Thus the pole values will have a small discrepancy.
- When using the ERP values from NEQ files they values are used for the exact times and everything is fully consistent.

When estimating the ERP parameters there are several subroutines that are being used all located in the module **DBcb_Data**:

1. **DBcb_SetEop** and **DBcb_FillEop:** These subroutines get the apriori ERP values for the reference time by linear interpolation of the values in the **DBcb_EOP** array. This subroutine is used only to initialize the ERP estimates.
2. **DBcb_PutEop:** This subroutine updates the value in the **DBcb_eop** array using the estimated values. On first call it replaces the original array with the new array aligned to the estimation interval times.
3. **DBcb_UpdEop**: Is used to update the values after the solution is generated.
4. **DBcb_ExtEop**: Is used to extrapolate the EOP file to cover the full “propagation” interval in case it is longer then the “determination” interval.

11.4 Using the ERP Estimates

11.4.1 Available ERP Output

Bahn outputs the ERP estimates in the asci parameter output file, and in the NEQ output files. Since the ERP parameters are a fundamental part of the terrestrial reference frame we will most likely always include them in the NEQ file from Bahn. UpdErp can convert the NEQ values to a proper NAPEOS ERP file and thus an additional/separate output file format does not seem to be necessary at present in NAPEOS.

The ERP files in IGS format, which are to be submitted with our orbit products, are generated by the NAPEOS program Par2Sinex. This program converts a Napeos NEQ file to the SINEX format and at the same time it can generate an ERP file in the format required for the IGS.

11.4.2 Orbit Fit and Orbit Prediction

When making an orbit fit through orbits given in the TRS it is of importance to use ERP information that is consistent with the orbit. The most critical factor is the UT1-UTC rate or LOD. But also the X-pole and Y-pole rates may cause significant effects. However, for NRT the knowledge of UT1-UTC is the most inaccurate one and therefore most prone to errors.

If we take our IGS Ultra-rapid products as example we have to make sure that the ERP information we use is taken from the same solution as the SP3 files that are used. This is done by the program **EopUpd** that puts together the ERP information just like **OrbUpd** puts together the SP3 information. **EopUpd** reads NEQ files and NAPEOS EOP files and merges the ERP information contained in those files and writes the output in the Napeos EOP file format.

The program **EopUpd** has the following abilities:

- Merge the ERP information form different NEQ files into a merged ERP file containing those ERP values which belong to the corresponding merged SP3 orbit file.
- Optionally: make the ERP values continuous at the “boundaries”. If making the ERPs continuous this means that basically only the drift of the parameters are used.
- In case of continuous ERP output the “reference” value should be defined.
- The program should get missing information (e.g. dX and dY) from the “normal” ERP input file.

Whether or not the ERP file should be continuous at the boundaries of the estimated ERP values or not depends on what the larger error source is. Is the reference frame realisation or the ERP estimation the dominant error source? Based on some experimentation it has become clear that the X- and Y-ERP values should be used as estimated. For UT1UTC and LOD it has become clear that LOD should be used but not UT1UTC. For UT1UTC a reference values should be selected, e.g. a well estimated valued from the NAPEOS EOP file and this value should be “integrated” using the LOD estimates.