

GNSS Networks in Algebraic Graph Theory

A. Lannes

CNRS/SUPELEC/Univ Paris-Sud* (France)

S. Gratton

CNES/OMP/DTP* (France)

Abstract. A new approach to GNSS networks is presented. Here, this approach is restricted to the case where the user handles the network data for his own objectives: the satellite-clock biases are not estimated. To deal with the general case where some data are missing, the corresponding theoretical framework appeals to some elementary notions of algebraic graph theory. As clarified in the paper, the notion of closure delay (CD) then generalizes that of double difference (DD). The body of the paper is devoted to the implications of this approach in GNSS data processing. One is then led to define local variables, which depend on the successive epochs of the time series, and a global variable which remains the same all over these epochs, with however possible state transitions from time to time. In the period defined by two successive transitions, the problem to be solved in the least-square sense is governed by a linear equation in which the key matrix has an angular block structure. This structure is well suited to recursive QR factorization. The state transitions induced by the variations of the GNSS graph are then handled in an optimal manner. Solving the integer-ambiguity problem via LLL decorrelation techniques is also made easier. At last but not the least, in centralized mode, this approach is particularly well suited to quality control.

Keywords. GNSS, DGPS, RTK. Centralized differences. Quality control, DIA. Ambiguity resolution, LLL.

1 Introduction

When processing times series of global positioning data, one is led to introduce 'local variables' u_k which depend on the successive epochs t_k of the time series to be processed, and a 'global variable' v which remains the same all over these epochs with however possible state transi-

tions from time to time. For example, the latter occur when some receiver-satellite signals appear or disappear. In the period defined by two successive transitions, the problem to be solved in the least-square (LS) sense is governed by a system of linear equations of the form

$$\begin{cases} A_1 u_1 + B_1 v = b_1 \\ A_2 u_2 + B_2 v = b_2 \\ \vdots \\ A_k u_k + B_k v = b_k \end{cases} \quad (1)$$

The definition of the variables u_k and v depends on the GNSS system under consideration. The components of u_k and v are real numbers, some components of v being integers: the integer ambiguities of the problem.

In matrix terms, Eq. (1) can be displayed as follows:

$$\begin{bmatrix} A_1 & & & B_1 \\ & A_2 & & B_2 \\ & & \ddots & \vdots \\ & & & A_k & B_k \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \\ \vdots \\ u_k \\ v \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_k \end{bmatrix} \quad (2)$$

As specified in Sect. 6.3 of Björck 1996 (see also Golub and van Loan 1989, Bierman 1977), the angular block structure of matrix $[A \ B]$ is well suited to recursive QR factorization. When dealing with large-scale problems, numerical accuracy can thereby be improved.

More interestingly, the corresponding techniques prove to be very efficient for GNSS data processing and quality control; see, e.g., Tiberius (1998), and Loehnert *et al* (2000). This is particularly the case for the GNSS centralized approaches (see Lannes and Gratton 2008). In particular, in the quality-control procedures, the identification of biases is then made easier. The approach presented in Lannes and Gratton (2008) was restricted to the simple case of continuous observations in RTK mode with a local-scale single baseline (see, e.g., Table 1 in Feng and Li 2008). The aim of the present paper is to extend this approach to the general case of the GNSS networks.

*This work was also supported by the CERFACS (France): the European Centre for Research and Advanced Training in Scientific Computation.

Other approaches have already been developed in this field. In particular, to raise the integer ambiguities in a simple manner, appropriate linear combinations of the original signals can be considered. The corresponding widelane techniques are very popular; see, e.g., Feng and Li (2008). This pointed out, when different approaches refer to the same physical models, the results must of course be the same. The best approach is then the most general and the most efficient. For example, with regard to integer-ambiguity resolution, the decorrelating properties of the widelane techniques are not optimal (see Teunissen, 1997). Likewise, the principle of the quality-control procedures must be well embedded in the theoretical framework of the selected approach.

2 Observational equations

The global positioning techniques are based on the following observational equations. For each frequency ν , for each receiver-satellite pair $(i, j) \equiv (r_i, s_j)$, and at each epoch t , the carrier-phase and code data are respectively of the form (see, e.g., Mercier and Laurichesse 2008)

$$\begin{aligned} \Phi_{\nu,t}(i, j) &= \rho_t(i, j) + T_t(i, j) - \alpha_\nu I_t(i, j) \\ &+ [f_{\phi;t}^{(r)}(i) - f_{\phi;t}^{(s)}(j) - \alpha_\nu [g_{\phi;t}^{(r)}(i) - g_{\phi;t}^{(s)}(j)] \\ &+ \lambda_\nu N_\nu(i, j) + \varepsilon_{\phi;\nu,t}(i, j) \end{aligned} \quad (3)$$

$$\begin{aligned} P_{\nu,t}(i, j) &= \rho_t(i, j) + T_t(i, j) + \alpha_\nu I_t(i, j) \\ &+ [f_{p;t}^{(r)}(i) - f_{p;t}^{(s)}(j) + \alpha_\nu [g_{p;t}^{(r)}(i) - g_{p;t}^{(s)}(j)] \\ &+ \varepsilon_{p;\nu,t}(i, j) \end{aligned} \quad (4)$$

In these equations, which are expressed in length units, $\rho_t(i, j)$ is the receiver-satellite range: the distance between satellite s_j (at the time $t - \tau$ where the signal is emitted) and receiver r_i (at the time t of its reception); $T_t(i, j)$ and $I_t(i, j)$ are the tropospheric and ionospheric delays, respectively. Here,

$$\alpha_\nu = \nu_1^2 / \nu^2 = \lambda_\nu^2 / \lambda_1^2 \quad (5)$$

The λ_ν 's denote the wavelengths of the carrier waves. Note that $\alpha_{\nu_1} = 1$. The integers $N_\nu(i, j)$ are the integer carrier-phase ambiguities: $N_\nu(i, j) \in \mathbb{Z}$.

The instrumental delays and clock errors that for a given epoch depend only on r_i are lumped together in the 'extended receiver-clock biases' $f_{\phi;t}^{(r)}(i)$, $f_{p;t}^{(r)}(i)$. Likewise, the instrumental delays and clock errors that for a given epoch depend only on s_j are lumped together in the 'extended satellite-clock biases' $f_{\phi;t}^{(s)}(j)$ and $f_{p;t}^{(s)}(j)$.

Similarly, $g_{\phi;t}^{(s)}(j)$, $g_{p;t}^{(r)}(i)$ and $g_{\phi;t}^{(s)}(j)$, $g_{p;t}^{(s)}(j)$ denote the biases induced by the time group delays.

In this model, which will be refined in Sect. 9 for quality control (see Eq. (112)), the expectation values of the noise terms $\varepsilon_{\phi;\nu,t}(i, j)$ and $\varepsilon_{p;\nu,t}(i, j)$ are supposed to be nought.

In this paper, we also assume that these noises are not mutually correlated.

A priori, on the grounds of Eqs. (3) and (4), two options are to be considered. In the first one, the extended satellite-clock biases $f_{\phi;t}^{(s)}(j)$ and $f_{p;t}^{(s)}(j)$ are not estimated. This option is well suited to a user who deals with the network data for his own objectives. In the second one, these biases are to be estimated; they are broadcasted to the network users for their precise point positioning (PPP). The present paper, which completes the original contribution of Lannes (2008), is devoted to the first option only. The second will be dealt with in a forthcoming contribution.

For our present purposes, we write the observational equations (3) and (4) in the form

$$\begin{aligned} \Phi_{\nu,t}(i, j) &= \rho_t(i, j) + T_t(i, j) - \alpha_\nu I_t(i, j) + \lambda_\nu N_\nu(i, j) \\ &+ \varphi_{\phi;\nu,t}^{(r)}(i) + \varphi_{\phi;\nu,t}^{(s)}(j) + \varepsilon_{\phi;\nu,t}(i, j) \end{aligned} \quad (6)$$

$$\begin{aligned} P_{\nu,t}(i, j) &= \rho_t(i, j) + T_t(i, j) + \alpha_\nu I_t(i, j) \\ &+ \varphi_{p;\nu,t}^{(r)}(i) + \varphi_{p;\nu,t}^{(s)}(j) + \varepsilon_{p;\nu,t}(i, j) \end{aligned} \quad (7)$$

where

$$\begin{cases} \varphi_{\phi;\nu,t}^{(r)}(i) &:= f_{\phi;t}^{(r)}(i) - \alpha_\nu g_{\phi;t}^{(r)}(i) \\ \varphi_{\phi;\nu,t}^{(s)}(j) &:= -[f_{\phi;t}^{(s)}(j) - \alpha_\nu g_{\phi;t}^{(s)}(j)] \end{cases} \quad (8)$$

and

$$\begin{cases} \varphi_{p;\nu,t}^{(r)}(i) &:= f_{p;t}^{(r)}(i) + \alpha_\nu g_{p;t}^{(r)}(i) \\ \varphi_{p;\nu,t}^{(s)}(j) &:= -[f_{p;t}^{(s)}(j) + \alpha_\nu g_{p;t}^{(s)}(j)] \end{cases} \quad (9)$$

3 Preliminary notions

We first introduce the notion of 'GNSS grid' and the related concept of 'GNSS graph' (Sect. 3.1). We then define the GNSS spaces to be considered (Sect. 3.2). The functions lying in these spaces can be decomposed in a 'differential manner.' The related notion is introduced in Sect. 3.3. The observational equations are then rewritten accordingly (Sect. (3.4)).

3.1 GNSS grid and GNSS graph

Let $\vartheta(i, j)$ be a function such as $\Phi_{\nu,t}(i, j)$ or $\rho_t(i, j)$ for example. Such a function takes its values on the points of a rectangular grid G_0 . When the GNSS device includes m receivers and n satellites, G_0 has m lines and n columns. More precisely, as some data may be missing, the values of ϑ are defined on n_e grid points with

$$n_e \leq mn \quad (10)$$

In the example presented in the upper part of Fig. 1, these points are surrounded by a small circle. They form

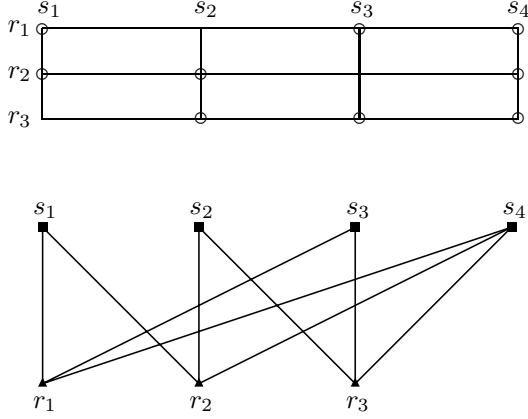


Figure 1: GNSS grid G and GNSS graph \mathcal{G} . In the example shown here, the GNSS graph includes 7 vertices (3 receivers and 4 satellites) and nine edges; $m = 3$, $n = 4$, $n_e = 9$. The data corresponding to the receiver-satellite pairs (r_1, s_2) , (r_2, s_3) and (r_3, s_1) are missing.

a subgrid of G_0 denoted by G : the GNSS grid. (Note that G implicitly depends on t .) As illustrated in the lower part of this figure, the points (i, j) of G correspond to the ‘edges’ (r_i, s_j) of the GNSS graph to be considered; \mathcal{E} denotes the set of its edges; n_e is their number. The receivers and the satellites involved in the definition of these edges define the ‘vertices’ of this graph; \mathcal{V} denotes the set of its vertices, and n_v their number:

$$n_v = m + n \quad (11)$$

A GNSS graph \mathcal{G} is therefore defined by the pair $(\mathcal{V}, \mathcal{E})$:

$$\mathcal{G} \equiv \mathcal{G}(\mathcal{V}, \mathcal{E})$$

Such a graph is connected (e.g., Biggs 1996): given any two vertices of \mathcal{V} , there exists a path of edges of \mathcal{E} connecting these vertices. When $n_e = mn$, the GNSS graph is said to be ‘full.’ Note that a full GNSS graph is not ‘complete:’ $mn < n_v(n_v - 1)/2$.

3.2 Edge-delay spaces

A function ϑ taking its values on G , and thereby on \mathcal{E} , can be regarded as a vector of the ‘edge-delay space’ $E \equiv \mathbb{R}^{n_e}$. The values of ϑ on G are then regarded as the components of ϑ in the standard basis of this space. The norm in E is therefore defined by the relation

$$\|\vartheta\|_E^2 = \sum_{(i,j) \in G} |\vartheta(i,j)|^2 \quad (12)$$

We now adopt the notation according which $\Psi_{\psi,t}$ stands for $\Phi_{\nu,t}$ if $\psi = (\phi; \nu)$, or for $P_{\nu,t}$ if $\psi = (p; \nu)$. The variance-covariance matrix of $\Psi_{\psi,t}$ is denoted by $[V_{\psi,t}]$. Let us then consider a function ϑ of type ψ , for example

a phase observational residual. At epoch t , the quadratic size of such a function is defined by the relation

$$\begin{aligned} \|\vartheta\|_{\psi,t}^2 &:= [\vartheta]^T [V_{\psi,t}]^{-1} [\vartheta] \\ &\equiv (\vartheta \cdot V_{\psi,t}^{-1} \vartheta) \end{aligned} \quad (13)$$

Here, $[\vartheta]$, is the column matrix whose entries are the components of ϑ on G ; (\cdot) is the inner product of the Euclidean space E . The space of functions ϑ with inner product

$$\langle \vartheta' | \vartheta \rangle_{\psi,t} := (\vartheta' \cdot V_{\psi,t}^{-1} \vartheta) \quad (14)$$

is denoted by $E_{\psi,t}$. This space can be referred to as the ‘edge-delay space’ of type ψ at epoch t .

Let us now introduce the following Cholesky factorization of the inverse of $[V_{\psi,t}]$:

$$[V_{\psi,t}]^{-1} = [U_{\psi,t}]^T [U_{\psi,t}] \quad (15)$$

Here, $[U_{\psi,t}]$ is an invertible upper-triangular matrix. Setting

$$\vartheta_{\psi,t}^E := U_{\psi,t} \vartheta \quad (16)$$

we have, from Eqs. (13) and (15),

$$\|\vartheta\|_{\psi,t}^2 = \|\vartheta_{\psi,t}^E\|_E^2 \quad (17)$$

3.3 Differential decomposition of the edge-delay functions

To introduce the reader to this notion, we first restrict ourselves to the special case where the GNSS graph is full: $\mathcal{G} = \mathcal{G}_0$ ($n_e = mn$). The extension to the general case derives from the analysis presented in Sect. 4.2.

In the special case under consideration (see Fig. 2), the notion of ‘single difference’ (SD) is associated with the following operation on G_0 :

$$\vartheta^{[\text{sd}]}(i,j) := \vartheta(i,j) - \vartheta(1,j) \quad (18)$$

Here, r_1 is the selected reference receiver: $\vartheta^{[\text{sd}]}$ vanishes on the first line of G_0 . Let s_1 now be the reference satellite. The notion of ‘double difference’ (DD) is then through the relation

$$\vartheta^{[\text{dd}]}(i,j) := \vartheta^{[\text{sd}]}(i,j) - \vartheta^{[\text{sd}]}(i,1) \quad (19)$$

By construction, $\vartheta^{[\text{dd}]}$ vanishes on the first line and on the first column of G_0 (see Fig. 2). From Eq. (18), we have

$$\begin{aligned} \vartheta^{[\text{dd}]}(i,j) &= [\vartheta(i,j) - \vartheta(1,j)] - [\vartheta(i,1) - \vartheta(1,1)] \\ &= \vartheta(i,j) - \vartheta(i,1) - [\vartheta(1,j) - \vartheta(1,1)] \end{aligned}$$

As a result, any edge-delay function ϑ can be decomposed in the form

$$\vartheta(i,j) = \vartheta^{[\text{r}]}(i) + \vartheta^{[\text{s}]}(j) + \vartheta^{[\text{dd}]}(i,j) \quad (20)$$

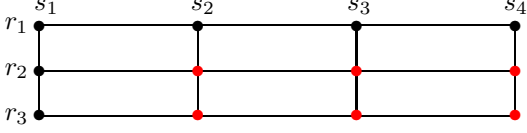


Figure 2: *Full GNSS grid.* In the special case where the GNSS graph is full, the double-difference function $\vartheta^{[\text{dd}]}$ is defined by its values on the $(m-1)(n-1)$ points shown here as red dots; see Eq. (19). By construction, $\vartheta^{[\text{dd}]}$ vanishes on the other points shown as black dots.

where

$$\vartheta^{[\text{r}]}(i) := \vartheta(i, 1) \quad (21)$$

and

$$\vartheta^{[\text{s}]}(j) := \vartheta(1, j) - \vartheta(1, 1) \quad (22)$$

As it is the case here, throughout this paper, the reference satellite, here s_1 , defines the origin of the receiver and satellite delays: $\vartheta^{[\text{s}]}(1) = 0$.

The integer-valued function $N_\nu(i, j)$, in particular, can therefore be decomposed in the form

$$N_\nu(i, j) = N_\nu^{[\text{r}]}(i) + N_\nu^{[\text{s}]}(j) + N_\nu^{[\text{dd}]}(i, j) \quad (23)$$

where

$$N_\nu^{[\text{r}]}(i) := N_\nu(i, 1) \quad (24)$$

and

$$N_\nu^{[\text{s}]}(j) := N_\nu(1, j) - N_\nu(1, 1) \quad (25)$$

3.4 Reference equations

According to Eq. (23), the phase equation (6) can be expanded in the form

$$\begin{aligned} \Phi_{\nu,t}(i, j) &= \rho_t(i, j) + T_t(i, j) - \alpha_\nu I_t(i, j) \\ &\quad + \lambda_\nu N_\nu^{[\text{dd}]}(i, j) \\ &\quad + \tilde{\varphi}_{\phi;\nu,t}^{[\text{r}]}(i) + \tilde{\varphi}_{\phi;\nu,t}^{[\text{s}]}(j) + \varepsilon_{\phi;\nu,t}(i, j) \end{aligned} \quad (26)$$

in which

$$\begin{cases} \tilde{\varphi}_{\phi;\nu,t}^{[\text{r}]}(i) &:= \varphi_{\phi;\nu,t}^{[\text{r}]}(i) + \varphi_{\phi;\nu,t}^{[\text{s}]}(1) \\ \tilde{\varphi}_{\phi;\nu,t}^{[\text{s}]}(j) &:= \varphi_{\phi;\nu,t}^{[\text{s}]}(j) - \varphi_{\phi;\nu,t}^{[\text{s}]}(1) \end{cases} \quad (27)$$

where

$$\begin{cases} \varphi_{\phi;\nu,t}^{[\text{r}]}(i) &:= \varphi_{\phi;\nu,t}^{(\text{r})}(i) + \lambda_\nu N_\nu^{[\text{r}]}(i) \\ \varphi_{\phi;\nu,t}^{[\text{s}]}(j) &:= \varphi_{\phi;\nu,t}^{(\text{s})}(j) + \lambda_\nu N_\nu^{[\text{s}]}(j) \end{cases} \quad (28)$$

Note that, by construction, $\tilde{\varphi}_{\phi;\nu,t}^{[\text{s}]}(1) = 0$. The quantities

$\tilde{\varphi}_{\phi;\nu,t}^{[\text{r}]}(i)$ and $\tilde{\varphi}_{\phi;\nu,t}^{[\text{s}]}(j)$ for $j \neq 1$ are then regarded as real variables without any physical interest. The integer variables are then the DD ambiguities $N_\nu^{[\text{dd}]}(i, j)$ for $i \neq 1$ and $j \neq 1$; see Fig. 2. The other variables, those induced by the terms ρ_t , T_t and I_t via the linearization of the problem, are introduced in Sect. 5.

The code equation (7) is then written in the form

$$\begin{aligned} P_{\nu,t}(i, j) &= \rho_t(i, j) + T_t(i, j) + \alpha_\nu I_t(i, j) \\ &\quad + \varphi_{p;\nu,t}^{[\text{r}]}(i) + \varphi_{p;\nu,t}^{[\text{s}]}(j) + \varepsilon_{p;\nu,t}(i, j) \end{aligned} \quad (29)$$

where (see (Eq. (9))

$$\begin{cases} \varphi_{p;\nu,t}^{[\text{r}]}(i) &:= \varphi_{p;\nu,t}^{(\text{r})}(i) + \varphi_{p;\nu,t}^{(\text{s})}(1) \\ \varphi_{p;\nu,t}^{[\text{s}]}(j) &:= \varphi_{p;\nu,t}^{(\text{s})}(j) - \varphi_{p;\nu,t}^{(\text{s})}(1) \end{cases} \quad (30)$$

Note that $\varphi_{p;\nu,t}^{[\text{s}]}(1) = 0$. The quantities $\varphi_{p;\nu,t}^{[\text{r}]}(i)$ and $\varphi_{p;\nu,t}^{[\text{s}]}(j)$ for $j \neq 1$ are other real variables of the problem.

4 Theoretical framework

We first introduce some elementary notions of algebraic graph theory (Sect. 4.1). We then establish the general properties underlying our approach (Sect. 4.2).

4.1 GNSS spanning tree and loops

As illustrated in Fig. 3, a spanning tree of $\mathcal{G} \equiv \mathcal{G}(\mathcal{V}, \mathcal{E})$ is a subgraph $\mathcal{G}_{\text{st}} \equiv \mathcal{G}(\mathcal{V}, \mathcal{E}_{\text{st}})$ formed by n_ν vertices and $n_\nu - 1$ edges, with no ‘cycle’ in it. Here, ‘cycle’ is used in the sense defined in algebraic graph theory (Biggs 1996). In the GNSS community, to avoid any confusion with the usual notion of wave cycle, it is not forbidden to substitute the term of ‘loop’ for that of ‘cycle.’ In this context, the number of loops defined through a given fixed (but arbitrary) spanning tree is the number of edges of \mathcal{E} that do not lie in \mathcal{E}_{st} . These edges,

$$c(q) := (r_{i(q)}, s_{j(q)}) \quad (31)$$

are said to be ‘loop-closure edges’ (see Fig. 3). Their number is denoted by n_c :

$$n_c = n_e - (n_\nu - 1) \quad (n_\nu = m + n, n_e \leq mn) \quad (32)$$

To select a GNSS spanning tree, the edges of \mathcal{E} are first ordered somehow. The corresponding sequence is of the form

$$e(q) := (r_{i_q}, s_{j_q}) \quad (q = 1, \dots, n_e)$$

The algorithm is then the following: set $q = 0$, $n_{\text{st}} = 0$, and $\mathcal{E}_{\text{st}} = \emptyset$ (the empty set). Then,

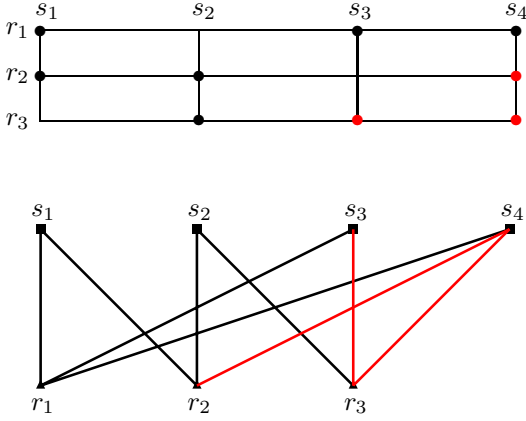


Figure 3: *GNSS spanning tree and loops.* The black edges of \mathcal{G} (the graph introduced in Fig. 1) are the edges of the selected spanning tree \mathcal{G}_{st} . The points of the corresponding subgrid G_{st} are shown as black dots. The remaining points of G (the red dots of G) correspond to the loop-closure edges (the red edges of \mathcal{G}). We then have one loop of order 4, and 2 loops of order 6: (r_2, s_4, r_1, s_1) , $(r_3, s_3, r_1, s_1, r_2, s_2)$ and $(r_3, s_4, r_1, s_1, r_2, s_2)$.

1. If $n_{\text{st}} = n_v - 1$, terminate the process; otherwise, set $q \stackrel{\text{set}}{=} q + 1$.
2. When the vertices of $e(q)$ are not connected via edges of \mathcal{E}_{st} , set $\mathcal{E}_{\text{st}} \stackrel{\text{set}}{=} \mathcal{E}_{\text{st}} \cup \{e(q)\}$ and $n_{\text{st}} \stackrel{\text{set}}{=} n_{\text{st}} + 1$; then go to step 1.

The subgrid of G corresponding to the edges of \mathcal{E}_{st} is denoted by G_{st} . By construction, the spanning tree thus found depends on how the edges are ordered.

Example 4.1. To show, in concrete manner, how this algorithm works, we now consider its action on the grid G of Fig. 3, its points being ordered line by line.

The points of the first line of G , the points $(1, 1)$, $(1, 3)$ and $(1, 4)$, define the first 3 edges of \mathcal{E}_{st} :

$$\mathcal{E}_{\text{st}} \stackrel{\text{set}}{=} \{(r_1, s_1), (r_1, s_3), (r_1, s_4)\} \quad (n_{\text{st}} = 3)$$

By construction, four vertices of \mathcal{G} are then connected: r_1, s_1, s_3 and s_4 .

The next point of G , the first point of line 2, is associated with edge (r_2, s_1) . As r_2 and s_1 are not connected via edges of \mathcal{E}_{st} , this edge cannot be not a loop-closure edge. We therefore set

$$\mathcal{E}_{\text{st}} \stackrel{\text{set}}{=} \mathcal{E}_{\text{st}} \cup \{(r_2, s_1)\} \quad (n_{\text{st}} = 4)$$

Five vertices are then connected: r_1, s_1, s_3, s_4 and r_2 .

The next point of line 2 is associated with edge (r_2, s_2) . As r_2 and s_2 are not connected via edges of \mathcal{E}_{st} , we set

$$\mathcal{E}_{\text{st}} \stackrel{\text{set}}{=} \mathcal{E}_{\text{st}} \cup \{(r_2, s_2)\} \quad (n_{\text{st}} = 5)$$

Six vertices are then connected: r_1, s_1, s_3, s_4, r_2 and s_2 .

The next point of G , the last point of line 2, is associated with edge (r_2, s_4) . As r_2 and s_4 are already connected, this edge closes a loop with some edges of \mathcal{E}_{st} . As a result, this edge is the first loop-closure edge: $c(1) = (r_2, s_4)$; see Eq. (24). The corresponding loop, (r_2, s_4, r_1, s_1) , is of order 4: it includes 4 edges (see Fig. 3).

The next point of G , the second point of line 3, is associated with edge (r_3, s_2) . As r_3 and s_2 are not connected via edges of \mathcal{E}_{st} , we then set

$$\mathcal{E}_{\text{st}} \stackrel{\text{set}}{=} \mathcal{E}_{\text{st}} \cup \{(r_3, s_2)\} \quad (n_{\text{st}} = 6)$$

As all the vertices of \mathcal{E} are then connected, the algorithm stops: \mathcal{E}_{st} is then completely defined.

The remaining points of line 3 therefore define two loop-closure edges: $c(2) = (r_3, s_3)$ and $c(3) = (r_3, s_4)$. These loops are of order 6; see Fig. 3.

Remark 4.1. In the special case of the graph shown in Fig. 3, there exist spanning trees for which the three loops are of order 4. In general, the choice of the spanning tree is arbitrary; see however Remark 4.2.

Remark 4.2. As explicitly shown in Sect. 7.4.3, to handle some ‘graph transitions,’ one is led to order the points of G in a more subtle manner. To write down the algorithm yielding the corresponding spanning tree, the reader is invited to build the spanning trees defined in the example given in that section (Example 8.1).

Remark 4.3. In the special case examined in Sect. 3.3, the GNSS graph is full: $\mathcal{G} = \mathcal{G}_0$. The points of G_{st} obtained by spanning G_0 line by line are then the n points of its first line, and the remaining $m - 1$ points of its first column (see Fig. 2). The other points, which form a subgrid with $m - 1$ lines and $n - 1$ columns, then correspond to loop-closure edges. All the loops are then of order four.

4.2 Reference properties

According to the properties established in this section, the analysis presented in Sect. 3.3 can be extended to the general case of GNSS networks with missing data. We first introduce the key notion of ‘vertex-delay space.’

Vertex-delay space. The functions of the form

$$\varphi(i, j) = \varphi^{[r]}(i) + \varphi^{[s]}(j) \quad (\text{with } \varphi^{[s]}(1) = 0) \quad (33)$$

form a subspace of the edge-delay space E . This subspace, denoted by F , can be referred to as the vertex-delay space. By definition, the ‘receiver-delay space’ $E^{[r]}$ is the space of functions $\varphi^{[r]}(i)$. Similarly, the ‘satellite-delay space’ $E^{[s]}$ is the space of functions $\varphi^{[s]}(j)$ such that $\varphi^{[s]}(1) = 0$. By construction, F is the direct sum of $E^{[r]}$ and $E^{[s]}$:

$$F = E^{[r]} + E^{[s]} \quad (34)$$

We thus have

$$\dim E^{[r]} = m \quad \dim E^{[s]} = n - 1 \quad (35)$$

and

$$\dim F = \dim E^{[r]} + \dim E^{[s]} = n_v - 1 \quad (36)$$

Property 1. *Given any edge-delay function ϑ taking its values on G , for each spanning tree \mathcal{G}_{st} of \mathcal{G} , there exists a unique set of receiver and satellite delays*

$$\Theta := \{\vartheta^{[r]}(i)\}_{i=1}^m \cup \{\vartheta^{[s]}(j)\}_{j=1}^n \quad \text{with} \quad \vartheta^{[s]}(1) = 0$$

such that $\vartheta(i, j) = \vartheta^{[r]}(i) + \vartheta^{[s]}(j)$ on the points of G_{st} .

More concretely, the following process provides these delays in a recursive manner.*

Recursive differential process. Set $\vartheta^{[s]}(1) = 0$; then, span the points of G_{st} line by line (see Fig. 3). For each point (i, j) thus encountered, then proceed as follows.

If $\vartheta^{[s]}(j)$ has already been fixed, and $\vartheta^{[r]}(i)$ is not fixed yet, set

$$\vartheta^{[r]}(i) = \vartheta(i, j) - \vartheta^{[s]}(j)$$

If $\vartheta^{[r]}(i)$ has already been fixed, and $\vartheta^{[s]}(j)$ is not fixed yet, set

$$\vartheta^{[s]}(j) = \vartheta(i, j) - \vartheta^{[r]}(i)$$

To obtain all these delays, G_{st} is to be spanned in this way as many times as required. The delay set Θ is unique. Indeed, applied to a function ϑ vanishing on the points of G_{st} , this recursive process provides nought delays.

It is important to point out that the only operations involved in this process are differences. As a result, if ϑ is an integer-valued function, the receiver and satellite delays $\vartheta^{[r]}(i)$ and $\vartheta^{[s]}(j)$ lie in \mathbb{Z} .

Example 4.2. To illustrate this recursive differential process, we now follow its action on the grid G_{st} of Fig. 3. As $\vartheta^{[s]}(1)$ is nought, we then obtain successively:

$$\vartheta^{[r]}(1) = \vartheta(1, 1) - \vartheta^{[s]}(1) = \vartheta(1, 1)$$

$$\vartheta^{[s]}(3) = \vartheta(1, 3) - \vartheta^{[r]}(1)$$

$$\vartheta^{[s]}(4) = \vartheta(1, 4) - \vartheta^{[r]}(1)$$

$$\vartheta^{[r]}(2) = \vartheta(2, 1) - \vartheta^{[s]}(1) = \vartheta(2, 1)$$

$$\vartheta^{[s]}(2) = \vartheta(2, 2) - \vartheta^{[r]}(2)$$

$$\vartheta^{[r]}(3) = \vartheta(3, 2) - \vartheta^{[s]}(2)$$

Closure delays. According to Property 1, the quantities

$$\vartheta^{[\text{cd}]}(i, j) := \vartheta(i, j) - [\vartheta^{[r]}(i) + \vartheta^{[s]}(j)] \quad (37)$$

vanish on the points of G_{st} . The values of $\vartheta^{[\text{cd}]}$ of interest are therefore defined on the remaining points of G , i.e., on the n_c loop-closure edges of \mathcal{G} (see Fig. 3 and Eq. (32)). These quantities can therefore be referred to as the ‘closure delays’ of ϑ , hence the notation cd or CD.

Remark 4.4. The notion of closure delay generalizes that of double difference; see Eq. (20), Figs. 2 and 3. In fact, the CD’s are algebraic sums of SD’s. For example, with regard to Example 4.2, the closure delay $\vartheta^{[\text{cd}]}(3, 4)$ can be displayed as follows (see Fig. 3):

$$[\vartheta(3, 4) - \vartheta(1, 4)] + [\vartheta(1, 1) - \vartheta(2, 1)] + [\vartheta(2, 2) - \vartheta(3, 2)]$$

Property 2. *Any edge-delay function ϑ taking its values on G can be decomposed in the form*

$$\vartheta(i, j) = \vartheta^{[r]}(i) + \vartheta^{[s]}(j) + \vartheta^{[\text{cd}]}(i, j)$$

For a given spanning tree, this decomposition is unique.

This property is a simple transcription of Eq. (37). The uniqueness of this decomposition results from Property 1.

Example 4.3. With regard to the GNSS grid of Fig. 3, let us consider (for simplicity) the ambiguity function

$$N: \begin{array}{|c|c|c|c|} \hline 2 & * & 1 & -1 \\ \hline -1 & 1 & * & 1 \\ \hline * & -2 & 2 & -1 \\ \hline \end{array}$$

The recursive differential process of Example 4.2 applied to this function yields the following components of N :

$$N^{[r]}: \begin{array}{|c|c|c|c|} \hline 2 & * & 2 & 2 \\ \hline -1 & -1 & * & -1 \\ \hline * & -4 & -4 & -4 \\ \hline \end{array}$$

$$N^{[s]}: \begin{array}{|c|c|c|c|} \hline 0 & * & -1 & -3 \\ \hline 0 & 2 & * & -3 \\ \hline * & 2 & -1 & -3 \\ \hline \end{array}$$

$$N^{[\text{cd}]}: \begin{array}{|c|c|c|c|} \hline 0 & * & 0 & 0 \\ \hline 0 & 0 & * & 5 \\ \hline * & 0 & 7 & 6 \\ \hline \end{array}$$

Closure-delay space. The functions ϑ that vanish on G_{st} form a subspace of E denoted by $E^{[\text{cd}]}$. This space is referred to as the ‘closure-delay space.’ For example, when \mathcal{G} is full, $E^{[\text{cd}]}$ is the corresponding DD space $E^{[\text{dd}]}$. From Eq. (32), we have

$$\dim E^{[\text{cd}]} = n_c \quad (38)$$

According to Property 2, E is the ‘oblique direct sum’ of $E^{[r]}$, $E^{[s]}$ and $E^{[\text{cd}]}$:

$$E = E^{[r]} + E^{[s]} + E^{[\text{cd}]} \quad (39)$$

*This type of recursive process was introduced for the first time in ‘phase-closure imaging;’ see Sect. 2E in Lannes (2005).

5 Statement of the problem

By taking account of Property 2, the phase equation (26) is then written in the form

$$\Phi_{\nu,t} = \rho_t + T_t - \alpha_\nu I_t + \lambda_\nu N_\nu^{[\text{cd}]} + \tilde{\varphi}_{\phi;\nu,t} + \varepsilon_{\phi;\nu,t} \quad (40)$$

where $\tilde{\varphi}_{\phi;\nu,t} := \tilde{\varphi}_{\phi;\nu,t}^{[r]} + \tilde{\varphi}_{\phi;\nu,t}^{[s]}$

Likewise, in terms of functions taking their values on grid G , the code equation (29) is now read as follows:

$$P_{\nu,t} = \rho_t + T_t + \alpha_\nu I_t + \varphi_{p;\nu,t} + \varepsilon_{p;\nu,t} \quad (41)$$

Here, $\varphi_{p;\nu,t} := \varphi_{p;\nu,t}^{[r]} + \varphi_{p;\nu,t}^{[s]}$

As specified in this section, the problem can then be stated in the terms of Eq. (1); see the context of that equation.

At this level, depending on the network geometry, some constraints may be introduced. For example, when the length of baseline $r_{i_1} \leftrightarrow r_{i_2}$ is sufficiently small, the constraints $T_t(i_1, j) = T_t(i_2, j)$ and $I_t(i_1, j) = I_t(i_2, j)$ are to be imposed. In most cases encountered in practice, the variables $\rho_t(i, j)$, $T_t(i, j)$ and $I_t(i, j)$ are linearly expanded in terms of other variables. In the general case, some of the latter depend on t , while others not; see, e.g., Feng and Li (2008). In other terms, the first ones are ‘local’ variables, while the others are ‘global’ with however possible transitions from time to time.

In the approach presented in this paper, the local variables $\tilde{\varphi}_{\phi;\nu,t}$ and $\varphi_{p;\nu,t}$ are regarded as particular variables of the problem. The other local variables, such as those involved in the linearization of ρ_t and T_t , are lumped together in some variable u_t . In general, the global variable v includes two blocks:

$$v = \begin{bmatrix} v_b \\ v_a \end{bmatrix} \quad (42)$$

The entries of v_a are the integer CD ambiguities $N_\nu^{[\text{cd}]}(i, j)$. The entries of v_b are simple real numbers. For example, when the position of some receivers and $I_t(i, j)$ are expanded as polynomial functions of t , the entries of v_b are the corresponding unknowns. This is also the case when orbital parameters are to be retrieved.

As is well known, in a first step (see Sect. 8 for the second), the integer variables are also dealt with as ‘float variables,’ i.e., as simple real variables.

Let us denote by $\bar{S}_k := \{s_1, s_2, \dots, s_{\bar{n}_k}\}$ the series of satellites involved in the process until epoch t_k included. A given satellite may disappear and reappear in the same run. Such a satellite is then regarded as a new satellite. In other words, whenever this occurs, a new satellite is added at the end of this series. The n_k satellites of epoch t_k form a subset S_k of \bar{S}_k : $n_k \leq \bar{n}_k$.

To introduce the reader to what is essential, we first restrict ourselves to the case where the GNSS graph \mathcal{G} does

not change in the current run $[t_1, \dots, t_\kappa, \dots, t_k]$: no state transition occurs in this interval. In this case, we of course have $n_k = \bar{n}_k$.

5.1 Optimization principle

In the context previously defined, the observational equations (40) and (41) therefore lead to equations of the form

$$\tilde{\Psi}_{\psi,\kappa} = \mathcal{A}_{\psi,\kappa} u_\kappa + \mathcal{B}_{\psi,\kappa} v + \varphi_{\psi,\kappa} + \varepsilon_{\psi,\kappa} \quad (43)$$

with $\varphi_{\psi,\kappa}$ in F . For clarity, κ stands for t_κ ; $\Psi_{\psi,\kappa}$ thus stands for $\Phi_{\nu,\kappa}$ if $\psi = (\phi; \nu)$, or for $P_{\nu,\kappa}$ if $\psi = (p; \nu)$; $\mathcal{A}_{\psi,\kappa}$ and $\mathcal{B}_{\psi,\kappa}$ are linear operators. The notation $\tilde{\Psi}_{\psi,\kappa}$ means that the zero-order terms of this linearization are taken into account; see, e.g., Eqs. (14), (17) and (18) in Lannes and Gratton (2008). Here, the variable $\varphi_{\psi,\kappa}$ corresponds to the quantities $\tilde{\varphi}_{\phi;\nu,t}$ and $\varphi_{p;\nu,t}$ of Eqs. (40) and (41), respectively.

The problem is to minimize the objective functional (see Eq. (13))

$$\begin{aligned} \mathfrak{F}(u_1, \dots, u_\kappa, \dots, u_k, v; \dots, \varphi_{\phi;\nu,\kappa}, \varphi_{p;\nu,\kappa}, \dots) \\ := \sum_{\kappa=1}^k \sum_{\nu} \sum_{\psi=(\phi;\nu)}^{\psi=(p;\nu)} \|\theta_\psi(u_\kappa, v) - \varphi_{\psi,\kappa}\|_{\psi,\kappa}^2 \end{aligned} \quad (44)$$

where, from Eq. (43),

$$\theta_\psi(u_\kappa, v) := \tilde{\Psi}_{\psi,\kappa} - (\mathcal{A}_{\psi,\kappa} u_\kappa + \mathcal{B}_{\psi,\kappa} v) \quad (45)$$

For each frequency, the sum in ψ includes two terms, a phase term and a code term, hence the notation adopted in Eq. (44).

In our approach, this minimization is performed in two steps. The first step is to minimize \mathfrak{F} in the variables $\varphi_{\phi;\nu,\kappa}$ and $\varphi_{p;\nu,\kappa}$ for $\kappa = 1, \dots, k$, and for each κ , for all ν . We now clarify this point.

Given any ϑ in E , in particular for $\theta_\psi(u_\kappa, v)$, let us set

$$\varphi_{\psi,\kappa}^o := \operatorname{argmin}_{\varphi \in F} \|\vartheta - \varphi\|_{\psi,\kappa}^2 \quad (46)$$

As illustrated in Fig. 4, $\varphi_{\psi,\kappa}^o$ is the point of F closest to ϑ , the distance being that induced by the norm defined on $E_{\psi,\kappa}$; $\varphi_{\psi,\kappa}^o$ is therefore the projection of ϑ on F in $E_{\psi,\kappa}$:

$$\varphi_{\psi,\kappa}^o := \mathcal{P}_{\psi,\kappa} \vartheta \quad (47)$$

Let us now denote by $\mathcal{P}'_{\psi,\kappa}$ the projection (operator) of $E_{\psi,\kappa}$ onto the orthogonal complement of F :

$$\mathcal{P}'_{\psi,\kappa} \vartheta := \vartheta - \mathcal{P}_{\psi,\kappa} \vartheta \quad (48)$$

From Eq. (44), the second step, the heart of the problem, is therefore to minimize the reduced functional

$$\begin{aligned} \mathfrak{F}^r(u_1, \dots, u_\kappa, \dots, u_k, v) \\ := \sum_{\kappa=1}^k \sum_{\nu} \sum_{\psi=(\phi;\nu)}^{\psi=(p;\nu)} \|\mathcal{P}'_{\psi,\kappa} \theta_\psi(u_\kappa, v)\|_{\psi,\kappa}^2 \end{aligned} \quad (49)$$

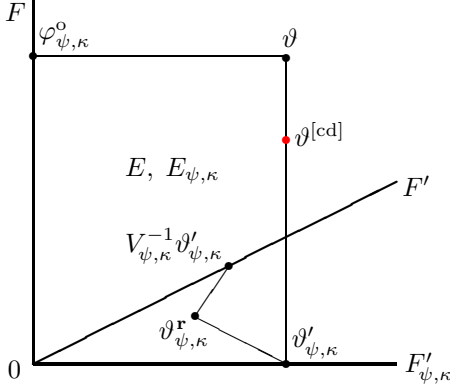


Figure 4: *Centralization and reduction.* In this geometrical illustration, $E_{\psi, \kappa}$ is the edge-delay space of type ψ at epoch t_{κ} (see Sect. 3.2); F is the vertex-delay space; $F'_{\psi, \kappa}$ is the orthogonal complement of F in $E_{\psi, \kappa}$, whereas F' is the orthogonal complement of F in the Euclidean space E . The functions lying in F' satisfy the ‘centralization property’ 4; $\varphi_{\psi, \kappa}^o$ is the orthogonal projection of ϑ on F in $E_{\psi, \kappa}$, whereas $\vartheta'_{\psi, \kappa}$ is the orthogonal projection of ϑ on $F'_{\psi, \kappa}$: $\vartheta'_{\psi, \kappa} = \vartheta - \varphi_{\psi, \kappa}^o$. According to Property 3, $V_{\psi, \kappa}^{-1} \vartheta'_{\psi, \kappa}$ lies in F' . By definition, $\vartheta_{\psi, \kappa}^r$ is equal to $U_{\psi, \kappa} \vartheta'_{\psi, \kappa}$ where $U_{\psi, \kappa}$ is defined via Eq. (15). The norm of $\vartheta'_{\psi, \kappa}$ in $E_{\psi, \kappa}$ is equal to that of $\vartheta_{\psi, \kappa}^r$ in E ; see Eqs. (56) and (12). As justified in Sect. 5.3, $\vartheta_{\psi, \kappa}^r$ is said to be the ‘ ψ -reduced form’ of ϑ . Note that $V_{\psi, \kappa}^{-1} \vartheta'_{\psi, \kappa} = U_{\psi, \kappa}^T \vartheta_{\psi, \kappa}^r$. In the special case where the variance-covariance matrix of $\Psi_{\psi, \kappa}$ is proportional to the identity, F' coincides with $F'_{\psi, \kappa}$.

5.2 Related properties

Denoting by F' the orthogonal complement of F in E , we then have the following property (see Fig. 4):

Property 3. *The vector $\varphi_{\psi, \kappa}^o$ is the vector φ of F for which $V_{\psi, \kappa}^{-1}(\vartheta - \varphi)$ lies in F' .*

Indeed, for any ξ in F , we have

$$\begin{aligned} \|\vartheta - (\varphi_{\psi, \kappa}^o + \xi)\|_{\psi, \kappa}^2 &= \|\vartheta - \varphi_{\psi, \kappa}^o\|_{\psi, \kappa}^2 + \|\xi\|_{\psi, \kappa}^2 \\ &\quad + 2\langle \xi | \vartheta - \varphi_{\psi, \kappa}^o \rangle_{\psi, \kappa} \end{aligned}$$

hence the property from Eqs. (46) and (14).

As specified below, the functions of F' satisfy particular ‘centralization properties.’

According to its definition (see Fig. 4), F' is the space of functions ϑ such that

$$\sum_{(i,j) \in G} \varphi(i,j) \vartheta(i,j) = 0 \quad (\text{for any } \varphi \text{ in } F) \quad (50)$$

From Eq. (33), the term on the left-hand side of this

equation can be expanded in the form

$$\begin{aligned} \sum_{(i,j) \in G} [\varphi^{[r]}(i) + \varphi^{[s]}(j)] \vartheta(i,j) \\ = \sum_{i=1}^m \varphi^{[r]}(i) \sum_{j \in \mathcal{L}_i} \vartheta(i,j) \\ + \sum_{j=2}^n \varphi^{[s]}(j) \sum_{i \in \mathcal{C}_j} \vartheta(i,j) \end{aligned}$$

Here, \mathcal{L}_i is the subset of G characterizing its i^{th} line:

$$\mathcal{L}_i := \{j : (i,j) \in G, i \text{ being fixed}\} \quad (51)$$

Likewise, \mathcal{C}_j is the subset of G characterizing its j^{th} column:

$$\mathcal{C}_j := \{i : (i,j) \in G, j \text{ being fixed}\} \quad (52)$$

The following property then results from Eq. (50):

Property 4. *The functions lying in F' satisfy the following conditions:*

$$\begin{cases} \sum_{j \in \mathcal{L}_i} \vartheta(i,j) = 0 & (\text{for } i = 1, \dots, m) \\ \sum_{i \in \mathcal{C}_j} \vartheta(i,j) = 0 & (\text{for } j = 2, \dots, n) \end{cases}$$

Note that the second condition then also holds for $j = 1$. For any ϑ in F' , we thus have

$$\begin{cases} \sum_{j \in \mathcal{L}_i} \vartheta(i,j) = 0 & (\text{for } i = 1, \dots, m) \\ \sum_{i \in \mathcal{C}_j} \vartheta(i,j) = 0 & (\text{for } j = 1, \dots, n) \end{cases}$$

In the special case where the GNSS graph is full, the lines and columns of G are also full. One then retrieves the characterization property of the ‘double-centralized functions’ of Shi and Han (1992):

$$\begin{cases} \sum_{j=1}^n \vartheta(i,j) = 0 & (\text{for } i = 1, \dots, m) \\ \sum_{i=1}^m \vartheta(i,j) = 0 & (\text{for } j = 1, \dots, n) \end{cases} \quad (53)$$

5.3 Reduced equations

According to Eqs. (17) and (16), we have, for any ϑ in E ,

$$\begin{aligned} \|\mathcal{P}'_{\psi, \kappa} \vartheta\|_{\psi, \kappa}^2 &= \|(\mathcal{P}'_{\psi, \kappa} \vartheta)_{\psi, \kappa}^E\|_E^2 \\ &= \|U_{\psi, \kappa} \mathcal{P}'_{\psi, \kappa} \vartheta\|_E^2 \end{aligned}$$

Setting (see Fig. 4)

$$\vartheta'_{\psi, \kappa} := \mathcal{P}'_{\psi, \kappa} \vartheta = \vartheta - \varphi_{\psi, \kappa}^o \quad (54)$$

and

$$\vartheta_{\psi,\kappa}^{\mathbf{r}} := U_{\psi,\kappa} \vartheta'_{\psi,\kappa} \quad (55)$$

we therefore have

$$\|\vartheta'_{\psi,\kappa}\|_{\psi,\kappa}^2 = \|\vartheta_{\psi,\kappa}^{\mathbf{r}}\|_E^2 \quad (56)$$

Note that

$$\vartheta_{\psi,\kappa}^{\mathbf{r}} = \mathcal{R}_{\psi,\kappa} \vartheta \quad (57)$$

where

$$\mathcal{R}_{\psi,\kappa} := U_{\psi,\kappa} \mathcal{P}'_{\psi,\kappa} \quad (58)$$

As the matrix elements of $U_{\psi,\kappa}$ are homogeneous to the inverse of a length (see Eq. (15)), $\vartheta_{\psi,\kappa}^{\mathbf{r}}$ is without any physical dimension. According to Eqs. (46), (54) and (56), the smallest value of $\|\vartheta - \varphi\|_{\psi,\kappa}$, φ spanning F , is equal to $\|\vartheta'_{\psi,\kappa}\|_{\psi,\kappa} = \|\vartheta_{\psi,\kappa}^{\mathbf{r}}\|_E$. For example, for any spanning tree, we have $\|\vartheta'_{\psi,\kappa}\|_{\psi,\kappa} < \|\vartheta^{[\text{cd}]}\|_{\psi,\kappa}$; see Fig. 4; $\vartheta_{\psi,\kappa}^{\mathbf{r}}$ can therefore be regarded as the ‘ ψ -reduced form of ϑ .’ Here, superscript \mathbf{r} stands for reduced. This pointed out, it can be shown that

$$\|\vartheta_{\psi,\kappa}^{\mathbf{r}}\|_E^2 = [\vartheta'_{\psi,\kappa}]^T [V'_{\psi,\kappa}]^{-1} [\vartheta'_{\psi,\kappa}] \quad (59)$$

where $[V'_{\psi,\kappa}]$ is the variance-covariance matrix of $\Psi'_{\psi,\kappa}$.

From Eqs. (49), (45), (56), (57) and (58), the reduced functional to be minimized is therefore of the form*

$$\begin{aligned} & \mathfrak{F}^{\mathbf{r}}(u_1, \dots, u_\kappa, \dots, u_k, v) \\ & := \sum_{\kappa=1}^k \sum_{\nu} \sum_{\psi=(\phi;\nu)}^{\psi=(p;\nu)} \|\tilde{\Psi}_{\psi,\kappa}^{\mathbf{r}} - (\mathcal{A}_{\psi,\kappa}^{\mathbf{r}} u_\kappa + \mathcal{B}_{\psi,\kappa}^{\mathbf{r}} v)\|_E^2 \end{aligned} \quad (60)$$

where

$$\mathcal{A}_{\psi,\kappa}^{\mathbf{r}} := \mathcal{R}_{\psi,\kappa} \mathcal{A}_{\psi,\kappa} \quad \mathcal{B}_{\psi,\kappa}^{\mathbf{r}} := \mathcal{R}_{\psi,\kappa} \mathcal{B}_{\psi,\kappa} \quad (61)$$

and

$$\tilde{\Psi}_{\psi,\kappa}^{\mathbf{r}} := \mathcal{R}_{\psi,\kappa} \tilde{\Psi}_{\psi,\kappa} \quad (62)$$

The reduced equations to be solved in the usual LS sense are therefore the following:

$$\begin{cases} \mathcal{A}_{\psi,1}^{\mathbf{r}} u_1 + \mathcal{B}_{\psi,1}^{\mathbf{r}} v = \tilde{\Psi}_{\psi,1}^{\mathbf{r}} \\ \vdots \\ \mathcal{A}_{\psi,\kappa}^{\mathbf{r}} u_\kappa + \mathcal{B}_{\psi,\kappa}^{\mathbf{r}} v = \tilde{\Psi}_{\psi,\kappa}^{\mathbf{r}} \\ \vdots \\ \mathcal{A}_{\psi,k}^{\mathbf{r}} u_k + \mathcal{B}_{\psi,k}^{\mathbf{r}} v = \tilde{\Psi}_{\psi,k}^{\mathbf{r}} \end{cases} \quad (63)$$

For each κ , we thus have an equation for $\psi = (\phi;\nu)$, and another one for $\psi = (p;\nu)$, and this for all the frequencies ν to be considered.

*This functional can equally well be obtained, directly, by considering the projection of Eq. (43) onto $F'_{\psi,\kappa}$. Indeed, Eq. (60) then derives from Eqs. (59) and (54).

In the dual-frequency case, for example, these equations can therefore be displayed in the block form (1) in which

$$A_k := \begin{pmatrix} \mathcal{A}_{\phi;\nu_1,k}^{\mathbf{r}} \\ \mathcal{A}_{\phi;\nu_2,k}^{\mathbf{r}} \\ \mathcal{A}_{p;\nu_1,k}^{\mathbf{r}} \\ \mathcal{A}_{p;\nu_2,k}^{\mathbf{r}} \end{pmatrix} \quad B_k := \begin{pmatrix} \mathcal{B}_{\phi;\nu_1,k}^{\mathbf{r}} \\ \mathcal{B}_{\phi;\nu_2,k}^{\mathbf{r}} \\ \mathcal{B}_{p;\nu_1,k}^{\mathbf{r}} \\ \mathcal{B}_{p;\nu_2,k}^{\mathbf{r}} \end{pmatrix} \quad (64)$$

and

$$b_k := \begin{pmatrix} \tilde{\Phi}_{\nu_1,k}^{\mathbf{r}} \\ \tilde{\Phi}_{\nu_2,k}^{\mathbf{r}} \\ \tilde{P}_{\nu_1,k}^{\mathbf{r}} \\ \tilde{P}_{\nu_2,k}^{\mathbf{r}} \end{pmatrix} \quad (65)$$

As clarified in Example 5.1 (Sect. 5.4), the entries of the matrices A_k , B_k and b_k can easily be computed.

5.4 Reference special case.

To illustrate our analysis in a concrete manner, we now consider the important special case where the variance-covariance matrix of the observational data $\Psi_{\psi,\kappa}$ is diagonal (see Liu 2002):

$$[V_{\psi,\kappa}] = \sigma_\psi^2 \text{diag}(\eta_\kappa(i, j)) \quad (\text{on } G) \quad (66)$$

Here, σ_ψ^2 is a ‘reference variance;’ $\eta(i, j)$ is a nonnegative weight function. Note that $U_{\psi,\kappa}$ is then defined by the relation

$$[U_{\psi,\kappa}] = \frac{1}{\sigma_\psi} \text{diag} \left(\frac{1}{\sqrt{\eta_\kappa(i, j)}} \right) \quad (\text{on } G) \quad (67)$$

From Eq. (46), $\varphi_{\psi,\kappa}^{\circ}$ then depends only on κ . For clarity, let us then set

$$\delta := \varphi_{\psi,\kappa}^{\circ} \quad \delta_{\mathbf{r},i} := \delta^{[\mathbf{r}]}(i) \quad \delta_{\mathbf{s},j} := \delta^{[\mathbf{s}]}(j) \quad (68)$$

and

$$\omega_\kappa(i, j) := \begin{cases} \frac{1}{\eta_\kappa(i, j)} & \text{on } G \\ 0 & \text{otherwise} \end{cases} \quad (69)$$

From Properties 3 and 4 (see also Eq. (33)), we then have

$$\begin{cases} \sum_{j \in \mathcal{L}_i} \omega_\kappa(i, j) \{ \vartheta(i, j) - [\delta_{\mathbf{r},i} + \delta_{\mathbf{s},j}] \} = 0 & (\text{for } i = 1, \dots, m) \\ \sum_{i \in \mathcal{C}_j} \omega_\kappa(i, j) \{ \vartheta(i, j) - [\delta_{\mathbf{r},i} + \delta_{\mathbf{s},j}] \} = 0 & (\text{for } j = 2, \dots, n) \end{cases}$$

i.e.,

$$\begin{cases} \sum_{j \in \mathcal{L}_i} \omega_{\kappa}(i, j) [\delta_{r,i} + \delta_{s,j}] = \sum_{j \in \mathcal{L}_i} \omega_{\kappa}(i, j) \vartheta(i, j) & (\text{for } i = 1, \dots, m) \\ \sum_{i \in \mathcal{C}_j} \omega_{\kappa}(i, j) [\delta_{r,i} + \delta_{s,j}] = \sum_{i \in \mathcal{C}_j} \omega_{\kappa}(i, j) \vartheta(i, j) & (\text{for } j = 2, \dots, n) \end{cases}$$

We are thus led to introduce the quantities

$$\begin{cases} \Omega_{r,i} := \sum_{j \in \mathcal{L}_i} \omega_{\kappa}(i, j) & (\text{for } i = 1, \dots, m) \\ \Omega_{s,j} := \sum_{i \in \mathcal{C}_j} \omega_{\kappa}(i, j) & (\text{for } j = 2, \dots, n) \end{cases}$$

and

$$\begin{cases} \theta_{r,i} := \sum_{j \in \mathcal{L}_i} \omega_{\kappa}(i, j) \vartheta(i, j) & (\text{for } i = 1, \dots, m) \\ \theta_{s,j} := \sum_{i \in \mathcal{C}_j} \omega_{\kappa}(i, j) \vartheta(i, j) & (\text{for } j = 2, \dots, n) \end{cases}$$

The equations to be solved to determine δ can then be written in the form

$$\begin{cases} \Omega_{r,i} \delta_{r,i} + \sum_{j=2}^n \Omega_{i,j} \delta_{s,j} = \theta_{r,i} & (\text{for } i = 1, \dots, m) \\ \sum_{i=1}^m \Omega_{i,j} \delta_{r,i} + \omega_{s,j} \delta_{s,j} = \theta_{s,j} & (\text{for } j = 2, \dots, n) \end{cases}$$

i.e., in matrix terms,

$$\begin{cases} [\Omega_r] [\delta_r] + [\Omega] [\delta_s] = [\theta_r] \\ [\Omega]^T [\delta_r] + [\Omega_s] [\delta_s] = [\theta_s] \end{cases} \quad (70)$$

Note that $[\Omega_r]$ is a $m \times m$ diagonal matrix, while $[\Omega_s]$ is a $(n-1) \times (n-1)$ diagonal matrix; their inverses are trivial. By construction, $[\Omega]$ has m lines and $n-1$ columns. As clarified below, Eq. (70) can be solved by computing the inverse of a matrix with size $(n-1) \times (n-1)$ or $m \times m$. We are thus led to consider two cases.

Case 1: $n-1 < m$

From the first equation (70), we have

$$[\delta_r] = [\Omega_r]^{-1} ([\theta_r] - [\Omega] [\delta_s]) \quad (71)$$

hence, from the second,

$$[\Omega]^T [\Omega_r]^{-1} ([\theta_r] - [\Omega] [\delta_s]) + [\Omega_s] [\delta_s] = [\theta_s]$$

i.e.,

$$[\tilde{\Omega}_s] [\delta_s] = [\theta_s] - [\Omega]^T [\Omega_r]^{-1} [\theta_r]$$

where $[\tilde{\Omega}_s]$ is the following $(n-1) \times (n-1)$ matrix:

$$[\tilde{\Omega}_s] := [\Omega_s] - [\Omega]^T [\Omega_r]^{-1} [\Omega] \quad (72)$$

It then follows that

$$[\delta_s] = [\tilde{\Omega}_s]^{-1} ([\theta_s] - [\Omega]^T [\Omega_r]^{-1} [\theta_r]) \quad (73)$$

Equation (71) then yields $[\delta_r]$.

Case 2: $n-1 \geq m$

From the second equation (70), we have

$$[\delta_s] = [\Omega_s]^{-1} ([\theta_s] - [\Omega]^T [\delta_r]) \quad (74)$$

hence, from the first,

$$[\Omega_r] [\delta_r] + [\Omega] [\Omega_s]^{-1} ([\theta_s] - [\Omega]^T [\delta_r]) = [\theta_r]$$

i.e.,

$$[\tilde{\Omega}_r] [\delta_r] = [\theta_r] - [\Omega] [\Omega_s]^{-1} [\theta_s]$$

where $[\tilde{\Omega}_r]$ is the following $m \times m$ matrix:

$$[\tilde{\Omega}_r] := [\Omega_r] - [\Omega] [\Omega_s]^{-1} [\Omega]^T \quad (75)$$

It then follows that

$$[\delta_r] = [\tilde{\Omega}_r]^{-1} ([\theta_r] - [\Omega] [\Omega_s]^{-1} [\theta_s]) \quad (76)$$

Equation (74) then yields $[\delta_s]$.

Example 5.1. In the special case defined in Fig. 3, let us concentrate on the column of B_k corresponding to the ambiguity variable $N_{\nu}^{[cd]}(2, 4)$. The entries of this column relative to the reduced data $\tilde{\Phi}_{\nu,k}^r(i, j)$ on G are then the corresponding values of $\lambda_{\nu}(\mathcal{R}_{\phi;\nu,k} c_{2,4})(i, j)$ where $c_{2,4}$ is the ‘characteristic function’ of edge (r_2, s_4) :

$$c_{2,4} := \begin{bmatrix} 0 & * & 0 & 0 \\ 0 & 0 & * & 1 \\ * & 0 & 0 & 0 \end{bmatrix}$$

The other entries of that column are nought. Let us then define the values of the weight function as follows (see Eq. (66)):

$$\eta_{\kappa} = \begin{bmatrix} 1.0 & * & 0.8 & 1.0 \\ 0.5 & 0.4 & * & 1.0 \\ * & 1.0 & 1.0 & 1.0 \end{bmatrix}$$

Equations (73) and (71) then provide $\delta \equiv \mathcal{P}_{\psi,k} c_{2,4}$:

$$\begin{aligned} \delta_{r,1} &= -0.22 & \delta_{r,2} &= 0.11 & \delta_{r,3} &= -0.21 \\ \delta_{s,2} &= -0.02 & \delta_{s,3} &= 0.22 & \delta_{s,4} &= 0.44 \end{aligned}$$

The values of the ψ -reduced form of $c_{2,4}$ for $\psi = (\phi; \nu)$ are then the following (see Eqs. (58), (48), (47), (67) and (68)):

$$\mathcal{R}_{\phi;\nu,k} c_{2,4} = \frac{1}{\sigma_{\phi;\nu}} \begin{bmatrix} 0.22 & * & 0.00 & -0.22 \\ -0.16 & -0.15 & * & 0.45 \\ * & 0.23 & 0.00 & -0.23 \end{bmatrix}$$

6 Solution of the problem: Survey

In the approach adopted in this paper, the equation (1) or (2) relative to the problem under consideration is solved in the LS sense, and recursively, by using the QR method (Sect. 7). The state transitions of the global variable, in particular those due to a change of the GNSS graph, are examined in that framework; see Sect. 7.4.

The selected QR implementation of this Recursive Least-Square (RLS) process is based on ‘Givens rotations’ (e.g., Björck 1996). The corresponding operations can thus be stored in memory very easily. The efficiency of the quality-control procedures is thereby increased; see Sect 9.

At each epoch t_k , the QR approach provides, in particular, the float ambiguity \hat{v}_a and the Cholesky factor $R_{k;a}$ of the inverse of its variance-covariance matrix. This upper-triangular matrix is then decorrelated via the LLL algorithm. As $R_{k;a}$ may be of large size, a particular implementation of this algorithm is proposed; see Sect. 8.2.

Once $R_{k;a}$ has thus been decorrelated, the integer ambiguity solution is obtained by using classical integer-programming techniques (see, e.g., Agrell et al. 2002). The problem can thereby be completely solved.

7 QR implementation

As already pointed out, we first restrict ourselves to the case where the GNSS graph \mathcal{G} does not change in the current run $[t_1, \dots, t_\kappa, \dots, t_k]$.

The notion of QR factorization is introduced in Sect. 7.1. We then show how to solve Eq. (2) in a recursive manner (Sect. 7.2). The corresponding variational aspects are presented in Sect. 7.3. We then specify how to handle the global variable when some transition occurs (Sect. 7.4).

7.1 QR factorization

Let us consider the following general LS problem: minimize, with the Euclidean norm,

$$\|Ax - y\|_{\mathbb{R}^m}^2 \quad (A \in \mathbb{R}^{m \times n}, m \geq n, \text{rank } A = n)$$

With regard to numerical accuracy, the best way to solve this problem is to use a method based on the QR factorization of A (see, e.g., Björck 1996):

$$A = Q \begin{bmatrix} R \\ 0 \end{bmatrix} \quad (77)$$

where $R \in \mathbb{R}^{n \times n}$ is an upper triangular matrix with positive diagonal terms, and $Q \in \mathbb{R}^{m \times m}$ is an orthogonal matrix: $Q^T Q = I_m$ (the identity matrix on \mathbb{R}^m). We thus

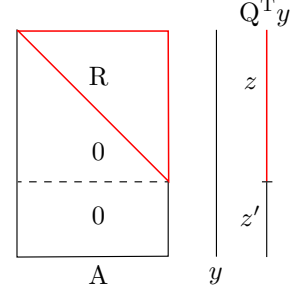


Figure 5: *LS solution via QR factorization.* The action of Q^T on A and y yields the basic QR structure sketched here: the upper-triangular matrix R and the column matrix z . The solution of the equation $Ax = y$ in the LS sense is then given by Eq. (78): $\hat{x} = R^{-1}z$.

have

$$\begin{aligned} \|Ax - y\|_{\mathbb{R}^m}^2 &= \|Q^T(Ax - y)\|_{\mathbb{R}^m}^2 \\ &= \left\| Q^T Q \begin{bmatrix} R \\ 0 \end{bmatrix} x - Q^T y \right\|_{\mathbb{R}^m}^2 \end{aligned}$$

Setting $Q^T y = z + z'$ where $z \in \mathbb{R}^n$ (see Fig. 5), it follows that

$$\|Ax - y\|_{\mathbb{R}^m}^2 = \|Rx - z\|_{\mathbb{R}^n}^2 + \|z'\|_{\mathbb{R}^{m-n}}^2$$

The LS solution is therefore given by the relation

$$\hat{x} = R^{-1}z \quad (78)$$

The problem can thereby be solved by back substitution. In the case where x is confined to \mathbb{Z}^n , the solution of the problem is therefore defined as follows:

$$\hat{x} = \underset{x \in \mathbb{Z}^n}{\operatorname{argmin}} \|R(x - \hat{x})\|_{\mathbb{R}^n}^2 \quad (79)$$

Indeed, $Rx - z = R(x - \hat{x})$.

According to Eq. (77), the QR factorization consists in finding an operator Q^T (and thereby an operator Q) such that $Q^T A$ has the block structure $[R \ 0]^T$ sketched in Fig. 5. This operator is defined as a product of elementary orthogonal transformations. In the implementation presented in this paper, the latter are Givens rotations (see Eqs. (2.3.10) to (2.3.13) in Björck 1996). Premultiplication of A and y by such a rotation matrix affects only rows k and ℓ of A and d . This matrix is defined so that, for $(a_k^2 + a_\ell^2) \neq 0$,

$$\begin{bmatrix} c & s \\ -s & c \end{bmatrix} \begin{bmatrix} a_k \\ a_\ell \end{bmatrix} = \begin{bmatrix} a \\ 0 \end{bmatrix} \quad (80)$$

where

$$a = (a_k^2 + a_\ell^2)^{1/2} \quad (81)$$

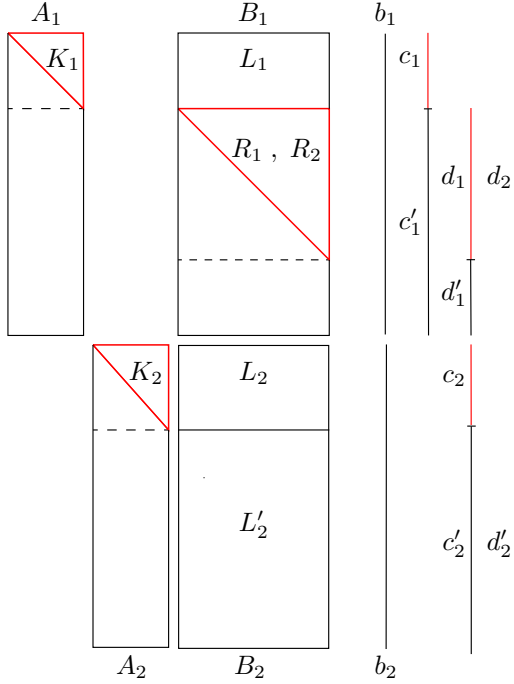


Figure 6: *LS solution via recursive QR factorization.* The principle of the recursive QR method is sketched here for the first two epochs: epoch 1 with the input block matrices A_1 , B_1 and the data column matrix b_1 ; epoch 2 with the input block matrices A_2 , B_2 and the data column matrix b_2 . The initialization process is performed in two steps: K_1 , (L_1, L'_1) , (c_1, c'_1) are built in the first step (see text for L'_1), whereas R_1 , (d_1, d'_1) are built in the second. The global float solution is then found by back substitution: $\hat{v} = R_1^{-1}d_1$. The local solution is then given by the formula $\hat{u}_1 = K_1^{-1}(c_1 - L_1\hat{v})$. Likewise, at the next epoch, one first builds K_2 , (L_2, L'_2) , (c_2, c'_2) , and then R_2 , (d_2, d'_2) ; \hat{v} is then updated via the relation $\hat{v} = R_2^{-1}d_2$. The local solution at epoch 2 can then be computed: $\hat{u}_2 = K_2^{-1}(c_2 - L_2\hat{v})$.

It is easy to check that the cosine and sinus values c and s are then given by the following formulas

$$c = a_k/a \quad s = a_\ell/a \quad (82)$$

Note that $m-1$ Givens rotations are required for the first column of A , $m-2$ for the second, and so on (see Fig. 5). It is important to point out that the action of Q^T can be stored in memory as the sequence of the successive (cosine, sinus) pairs (c, s) characterizing the successive Givens rotations involved in this operation.

7.2 Recursive QR factorization

We now show how to solve, in the LS sense and recursively, the equation (2) induced by the reduced equations.

Let us first consider the initialization epoch: epoch 1. The problem is then solved in two steps (see Fig. 6).

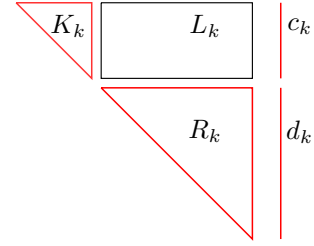


Figure 7: *Recursive QR triangular structure.* According to the principle of the recursive QR method sketched in Fig. 6, the calculation of R_{k+1} and d_{k+1} requires to have kept in memory the upper triangular matrix R_k and the column matrix d_k (see text).

The Givens rotations of the first step are those required for finding the upper triangular matrix K_1 . The modified version of B_1 thus obtained includes an upper block L_1 and a lower block L'_1 . Likewise, the modified version of b_1 includes two column submatrices: c_1 and c'_1 .

The Givens rotations of the second step yield the upper triangular matrix R_1 ; c'_1 then yields (d_1, d'_1) ; see Fig. 6. Note that K_1 , L_1 and c_1 are not affected by these rotations. The global solution is then obtained by back substitution via the formula $\hat{v} = R_1^{-1}d_1$. The local solution can then be also computed by back substitution: $\hat{u}_1 = K_1^{-1}(c_1 - L_1\hat{v})$.

The first step of the next epoch (epoch 2) is similar to that of epoch 1: one thus obtains the upper triangular matrix K_2 . The modified version of B_2 then includes an upper block L_2 and a lower block L'_2 . Likewise, the modified version of b_2 includes two column submatrices: c_2 and c'_2 (see Fig. 6). The Givens rotations of the second step then operate on (R_1, L'_2) and (d_1, c'_2) so as to transform L'_2 into a zero block matrix. One thus gets R_2 and (d_2, d'_2) ; \hat{v} is then updated via the relation $\hat{v} = R_2^{-1}d_2$. The local solution at epoch 2 can then be computed: $\hat{u}_2 = K_2^{-1}(c_2 - L_2\hat{v})$.

In summary, one thus operates, recursively, with the key structure shown in Fig. 7: K_{k+1} , (L_{k+1}, L'_{k+1}) and (c_{k+1}, c'_{k+1}) are computed from A_{k+1} , B_{k+1} and b_{k+1} , the quantities R_{k+1} and (d_{k+1}, d'_{k+1}) being then computed from (R_k, L'_{k+1}) and (d_k, c'_{k+1}) . We then have

$$\begin{bmatrix} K_{k+1} & L_{k+1} \\ \cdot & R_{k+1} \end{bmatrix} \begin{bmatrix} \hat{u}_{k+1} \\ \hat{v} \end{bmatrix} = \begin{bmatrix} c_{k+1} \\ d_{k+1} \end{bmatrix} \quad (83)$$

hence

$$\hat{v} = R_{k+1}^{-1}d_{k+1}$$

and

$$\hat{u}_{k+1} = K_{k+1}^{-1}(c_{k+1} - L_{k+1}\hat{v})$$

7.3 Variational calculation

We now answer to the following question: what are the variations $\Delta\hat{u}_k$ and $\Delta\hat{v}$ induced by a variation Δb_k of b_k (at epoch t_k)? From Eq. (2), these variations are the u - v components at epoch t_k of the LS solution of the equation

$$\begin{bmatrix} A_1 & & & & B_1 \\ & A_2 & & & B_2 \\ & & \ddots & & \vdots \\ & & & A_k & B_k \end{bmatrix} \begin{bmatrix} \Delta u_1 \\ \Delta u_2 \\ \vdots \\ \Delta u_k \\ \Delta v \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ \Delta b_k \end{bmatrix}$$

By construction, the quantities $\Delta d_1, \dots, \Delta d_{k-1}$ induced by this equation are nought. The problem is therefore the same as previously, Δd_k being then computed from $\Delta c'_k$ with $\Delta d_{k-1} = 0$. This is why it is recommended to store in memory the sequence of the successive pairs (c, s) characterizing the Givens operators involved in the two QR steps of epoch t_k (see Fig. 6 and Eqs. (81) & (82)).

7.4 State transitions of the global variable

At some epochs t_k , one may be led to perform a linear operation on the global variable v :

$$\hat{v}' = S\hat{v}$$

For example, this occurs in the following cases:

- 1) The components of \hat{v} are to be modified in a reversible manner; S is invertible: $S \equiv S_1$ (Sect. 7.4.1). Note that reordering the components of \hat{v} comes under this case.
- 2) Some components of \hat{v} are to be discarded; S is then of the form $S_2 S_1$ where (Sect. 7.4.2)
 - S_1 is a ‘reordering operator;’
 - S_2 is a ‘truncation operator.’

Such an operator is not invertible.

- 3) Some edges of the current spanning tree of \mathcal{G}_k (the GNSS graph at epoch t_k) are missing in \mathcal{G}_{k+1} . As specified in Sect. 7.4.3, S is then of the form $S_2 S_1$ where
 - S_1 is an operator which changes the set of the current CD ambiguities into another set of such ambiguities;
 - S_2 is a truncation operator.

At epoch t_{k+1} , new entries of v may appear. For example, this is the case when new edges appear in the GNSS graph. How to proceed in this case is specified in Sect. 7.4.4.

7.4.1 Reversible operations

We then have $R_k \hat{v} = R_k S_1^{-1} \hat{v}' = d_k$, hence

$$R'_k \hat{v}' = d_k \quad \text{where} \quad R'_k := R_k S_1^{-1}$$

This matrix is no longer triangular. One then performs Givens rotations on R'_k and d_k so that R'_k becomes upper triangular: $R'_k \rightarrow R''_k$, $d_k \rightarrow d''_k$. One then finally sets

$$R_k \stackrel{\text{set}}{=} R''_k \quad d_k \stackrel{\text{set}}{=} d''_k$$

7.4.2 Truncations

For example, consider the case where the components $\hat{v}^{(3)}$ and $\hat{v}^{(5)}$ of \hat{v} are to be discarded. One first performs the permutation

$$S_1 \begin{bmatrix} \hat{v}^{(1)} \\ \hat{v}^{(2)} \\ \hat{v}^{(3)} \\ \hat{v}^{(4)} \\ \hat{v}^{(5)} \\ \hat{v}^{(6)} \\ \vdots \end{bmatrix} = \begin{bmatrix} \hat{v}^{(3)} \\ \hat{v}^{(5)} \\ \hat{v}^{(1)} \\ \hat{v}^{(2)} \\ \hat{v}^{(4)} \\ \hat{v}^{(6)} \\ \vdots \end{bmatrix}$$

The columns of R_k are then permuted accordingly. As the matrix thus obtained, R'_k , is no longer upper triangular, one then performs Givens rotations on R'_k and d_k so that R'_k becomes upper triangular: $R'_k \rightarrow R''_k$, $d_k \rightarrow d''_k$. To complete the process, one then removes the first two lines and first two columns of R''_k , as well as the first two entries of \hat{v}' and d''_k . Again, one then finally sets $R_k \stackrel{\text{set}}{=} R''_k$ and $d_k \stackrel{\text{set}}{=} d''_k$.

7.4.3 Graph transitions

Let us denote by

$$\mathcal{G}_k := (\mathcal{V}_k, \mathcal{E}_k) \quad \mathcal{G}_{k+1} := (\mathcal{V}_{k+1}, \mathcal{E}_{k+1})$$

the GNSS graphs at epochs t_k and t_{k+1} , respectively. As illustrated in the upper and lower parts of Fig. 8, \mathcal{G}_k and \mathcal{G}_{k+1} denote their respective grids. Let us now set

$$\mathcal{F} := \mathcal{E}_k \cap \mathcal{E}_{k+1}$$

The edges of \mathcal{E}_k that do not lie in \mathcal{F} form a set denoted by \mathcal{M} :

$$\mathcal{M} := \mathcal{E}_k - \mathcal{F}$$

Likewise, the edges of \mathcal{E}_{k+1} that do not lie in \mathcal{F} form a set denoted by \mathcal{N} :

$$\mathcal{N} := \mathcal{E}_{k+1} - \mathcal{F}$$

Here, \mathcal{M} and \mathcal{N} stand for ‘missing edges’ and ‘new edges,’ respectively. In this section, we consider the case where \mathcal{M} is not empty.

The entries of \hat{v}_a at epoch t_k are defined with respect to the current spanning tree of \mathcal{G}_k : $\mathcal{G}_{k;st}$ (see Fig. 8). One is then led to introduce another spanning tree of \mathcal{G}_k : the ‘transition spanning tree’ $\mathcal{G}_{k;tst}$. This spanning tree is obtained by considering the following ‘ordered partition’ of \mathcal{E}_k (see Remark 4.2 and Fig. 8):

$$\mathcal{E}_k = \mathcal{F} \cup \mathcal{M} \quad (\mathcal{F} \cap \mathcal{M} = \emptyset)$$

The spanning tree of \mathcal{G}_{k+1} , $\mathcal{G}_{k+1;st}$, is then built from the following ‘ordered partition’ of \mathcal{E}_{k+1} (see Fig. 8):

$$\mathcal{E}_{k+1} = \mathcal{F} \cup \mathcal{N} \quad (\mathcal{F} \cap \mathcal{N} = \emptyset)$$

The analysis of the case where \mathcal{N} is not empty is completed under the analysis developed in Sect. 7.4.4.

With regard to \mathcal{M} , two cases are then to be considered:

- (i) the case where $\mathcal{M} \cap \mathcal{G}_{k;st} = \emptyset$;
- (ii) the case where $\mathcal{M} \cap \mathcal{G}_{k;st} \neq \emptyset$.

In the usual case (i), the elements of \mathcal{M} are loop-closure edges. The components of \hat{v}_a corresponding to these edges are then simply to be removed. The procedure described in Sect. 7.4.2 can then be implemented directly.

In the special case (ii), this is not so simple. One must first introduce some transition state $\hat{v}_{a;1}$. As specified in the example presented further on (see Fig. 8), the entries of $\hat{v}_{a;1}$ are defined with respect to the transition spanning tree $\mathcal{G}_{k;tst}$. More precisely, this state is defined by the relation $\hat{v}_{a;1} = S_1 \hat{v}_a$ where S_1 is an invertible operator defined via the recursive differential process defined in Sect. 4.2; R_k is then updated accordingly (see Sect. 7.4.1). This transition operation is completed by a truncation operation in which the components of $\hat{v}_{a;1}$ corresponding to the edges of \mathcal{M} are removed (see the procedure described in Sect. 7.4.2).

Once all these algebraic operations have been performed, the quantities R_k and d_k , which have thus been updated, are used to perform the recursive QR step towards R_{k+1} and d_{k+1} (see Sect. 7.4.4).

Example 7.1. To illustrate these considerations in a concrete manner, let us assume that the GNSS graph at epoch t_k is that shown in Fig. 3; its grid G_k is shown in the upper part of Fig. 8. The current spanning tree $\mathcal{G}_{k;st}$ is that represented in the same figure.

As shown in the lower grid of Fig. 8, let us now assume that at epoch t_{k+1} , satellite s_1 is no longer visible, and that satellite s_3 is no longer visible from receiver 1. A new satellite, s_5 , is then visible from receivers r_1 and r_2 . Moreover, satellite s_2 is then visible from receiver 1, and satel-

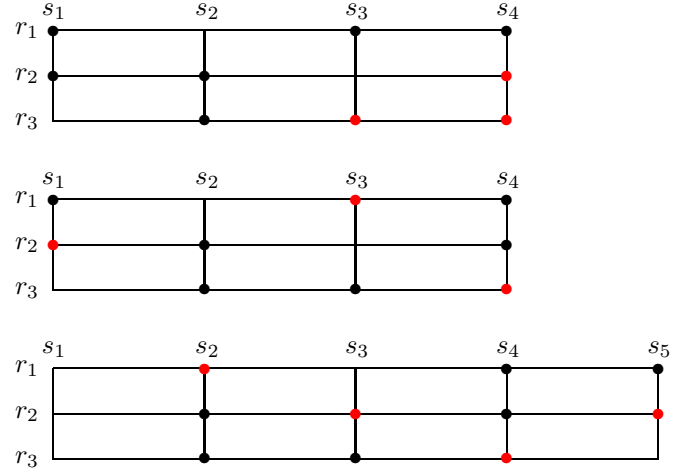


Figure 8: *Graph transition.* In the example shown here, the upper grid G_k is that of the GNSS graph at epoch t_k . The lower grid G_{k+1} is that of the GNSS graph at epoch t_{k+1} . The grid in between is a copy of G_k for describing the transition to be performed. As illustrated here, the edges (r_1, s_1) , (r_1, s_3) and (r_2, s_1) disappear at epoch t_{k+1} , while four new edges then appear: (r_1, s_2) , (r_1, s_5) , (r_2, s_3) and (r_2, s_5) . The black dots of the upper grid form the subgrid $G_{k;st}$ of the current spanning tree at that epoch. Likewise, the black dots of the second grid form the subgrid $G_{k;tst}$ of the transition spanning tree (see text). The red dots of each of these grids define the sets of loop-closure edges involved in the transition to be performed. The relation between the first ambiguity set and the second is linear and invertible. The corresponding operations can be performed by referring to the recursive process defined in Sect. 4.2. The black dots of the lower grid define the subgrid $G_{k+1;st}$ of the spanning tree selected for \mathcal{G}_{k+1} (for further details see text).

lite s_3 is then visible from receiver 2. We then have

$$\begin{aligned} \mathcal{F} &= \{(r_1, s_4), (r_2, s_2), (r_2, s_4), (r_3, s_2), (r_3, s_3), (r_3, s_4)\} \\ \mathcal{M} &= \{(r_1, s_1), (r_1, s_3), (r_2, s_1)\} \\ \mathcal{N} &= \{(r_1, s_2), (r_1, s_5), (r_2, s_3), (r_2, s_5)\} \end{aligned}$$

The grid points of $G_{k;st}$, $G_{k;tst}$ and $G_{k+1;st}$ are shown in Fig. 8 as black dots. The red dots correspond to the loop-closure edges defined via the choice of these spanning trees. The edges of $\mathcal{G}_{k;tst}$ and $\mathcal{G}_{k+1;st}$ are respectively obtained in the following orders:

$$\begin{aligned} &(r_1, s_4), (r_2, s_2), (r_2, s_4), (r_3, s_2), (r_3, s_3); (r_1, s_1) \\ &(r_1, s_4), (r_2, s_2), (r_2, s_4), (r_3, s_2), (r_3, s_3); (r_1, s_5) \end{aligned}$$

The entries of \hat{v}_a at epoch t_k are then the float CD ambiguities (see the upper grid of Fig. 8)

$$\hat{N}_\nu^{[cd]}(2, 4), \hat{N}_\nu^{[cd]}(3, 3), \hat{N}_\nu^{[cd]}(3, 4)$$

The entries of $\hat{v}_{a;1}$ at epoch t_k are then the float CD ambiguities (see the second grid of Fig. 8)

$$\hat{N}_{\nu;1}^{[cd]}(1, 3), \hat{N}_{\nu;1}^{[cd]}(2, 1), \hat{N}_{\nu;1}^{[cd]}(3, 4)$$

To define this state transition, i.e., to define, explicitly, the operator S_1 involved in the relation $\hat{v}_{a;1} = S_1 \hat{v}_a$, we are then led to consider the function $\vartheta \triangleq \hat{N}_\nu^{\text{set}} \hat{N}_\nu^{\text{[cd]}}$. Note that we then have

$$\begin{aligned} \vartheta(1,1) &= 0; & \vartheta(1,3) &= 0; & \vartheta(1,4) &= 0; \\ \vartheta(2,1) &= 0; & \vartheta(2,2) &= 0; & \vartheta(2,4) &= \hat{N}_\nu^{\text{[cd]}}(2,4); \\ \vartheta(3,2) &= 0; & \vartheta(3,3) &= \hat{N}_\nu^{\text{[cd]}}(3,3); & \vartheta(3,4) &= \hat{N}_\nu^{\text{[cd]}}(3,4) \end{aligned}$$

Conducted on grid $G_{k;\text{tst}}$ with $\vartheta^{[\text{s}]}(2) = 0$, and applied to this function, the recursive differential process defined Sect. 4.2 then yields, successively,

$$\begin{aligned} \vartheta^{[\text{r}]}(2) &= \vartheta(2,2) - \vartheta^{[\text{s}]}(2) = 0 \\ \vartheta^{[\text{s}]}(4) &= \vartheta(2,4) - \vartheta^{[\text{r}]}(2) = \hat{N}_\nu^{\text{[cd]}}(2,4) \\ \vartheta^{[\text{r}]}(3) &= \vartheta(3,2) - \vartheta^{[\text{s}]}(2) = 0 \\ \vartheta^{[\text{s}]}(3) &= \vartheta(3,3) - \vartheta^{[\text{r}]}(3) = \hat{N}_\nu^{\text{[cd]}}(3,3) \\ \vartheta^{[\text{r}]}(1) &= \vartheta(1,4) - \vartheta^{[\text{s}]}(4) = -\hat{N}_\nu^{\text{[cd]}}(2,4) \\ \vartheta^{[\text{s}]}(1) &= \vartheta(1,1) - \vartheta^{[\text{r}]}(1) = \hat{N}_\nu^{\text{[cd]}}(2,4) \end{aligned}$$

hence, from Eq. (37),

$$\begin{aligned} \hat{N}_{\nu;1}^{\text{[cd]}}(1,3) &= \vartheta(1,3) - [\vartheta^{[\text{r}]}(1) + \vartheta^{[\text{s}]}(3)] \\ &= \hat{N}_\nu^{\text{[cd]}}(2,4) - \hat{N}_\nu^{\text{[cd]}}(3,3) \\ \hat{N}_{\nu;1}^{\text{[cd]}}(2,1) &= \vartheta(2,1) - [\vartheta^{[\text{r}]}(2) + \vartheta^{[\text{s}]}(1)] \\ &= -\hat{N}_\nu^{\text{[cd]}}(2,4) \\ \hat{N}_{\nu;1}^{\text{[cd]}}(3,4) &= \vartheta(3,4) - [\vartheta^{[\text{r}]}(3) + \vartheta^{[\text{s}]}(4)] \\ &= \hat{N}_\nu^{\text{[cd]}}(3,4) - \hat{N}_\nu^{\text{[cd]}}(2,4) \end{aligned}$$

The operator S_1 involved in this reversible transition is thus explicitly defined; see Sect. 7.4.1. Note that its matrix and its inverse can be obtained via elementary algebraic computations. The ambiguities $\hat{N}_{\nu;1}^{\text{[cd]}}(1,3)$ and $\hat{N}_{\nu;1}^{\text{[cd]}}(2,1)$ are then discarded; see Sect. 7.4.2.

7.4.4 Handling additional components

From time to time, some new entries of v are to be introduced. For example, in the graph transition of Fig. 8, the following entries of v_a must be taken into account:

$$N_\nu^{\text{[cd]}}(1,2), N_\nu^{\text{[cd]}}(2,3), N_\nu^{\text{[cd]}}(2,5)$$

The first columns of B_{k+1} are then processed as the last columns of A_{k+1} (see Fig. 6). To get R_{k+1} and d_{k+1} , one then proceeds as illustrated in Fig. 9.

This pointed out, when such a transition occurs, one may be led to reorder the components of v ; see for instance Eq. (42). One then proceeds as specified in Sect. 7.4.1.

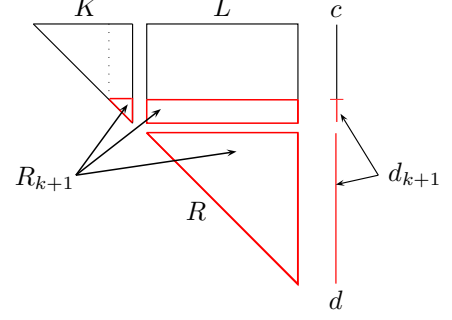


Figure 9: Handling additional entries of the global variable. When new entries of v appear at epoch t_{k+1} , the first columns of B_{k+1} are processed as the last columns of A_{k+1} (see Fig. 6). The recursive QR operation then yields the quantities K , L , c , R and d . To get R_{k+1} and d_{k+1} , one then proceeds as illustrated here.

8 Integer-ambiguity resolution

At each epoch, the QR approach provides, in particular, the float solution \hat{v} and the Cholesky factor R_k of the inverse of its variance-covariance matrix. We then have $R_k \hat{v} = d_k$ (see Sect. 7.2), i.e., from Eq. (42),

$$\begin{bmatrix} R_{k;b} & R_{k;ba} \\ 0 & R_{k;a} \end{bmatrix} \begin{bmatrix} \hat{v}_b \\ \hat{v}_a \end{bmatrix} = \begin{bmatrix} d_{k;b} \\ d_{k;a} \end{bmatrix} \quad (84)$$

The ambiguity solution is then defined by the relation (see Eq. (79))

$$\hat{v}_a = \operatorname{argmin}_{v_a \in \mathbb{Z}^{n_a}} \|R_{k;a}(v_a - \hat{v}_a)\|_{\mathbb{R}^{n_a}}^2 \quad (85)$$

where n_a is the number of entries of \hat{v}_a . When in the data assimilation process, \hat{v}_a becomes consistent with the model (up to the noise), the ambiguities are said to be fixed. The estimate of the float component of the global variable is then refined accordingly (see Eq. (84)):

$$\hat{v}_b = R_{k;b}^{-1}(d_{k;b} - R_{k;ba} \hat{v}_a) \quad (86)$$

The local variable \hat{u}_k is then refined via a FLS (Fixed Least-Squares) process, i.e., a process in which the ambiguities are fixed. Again, the QR method is well suited to solving this problem.

The remainder of this section is devoted to the search of the integer ambiguity solution. For clarity, subscripts ‘ k ’ and ‘ a ’ are then omitted. Equation (85), for instance, is then simply read as follows:

$$\hat{v} = \operatorname{argmin}_{v \in \mathbb{Z}^n} \|R(v - \hat{v})\|_{\mathbb{R}^n}^2 \quad (87)$$

This nearest-lattice-point problem is solved in two steps (see, e.g., Agrell et al. 2002). One first searches a ‘reduced basis’ of \mathbb{Z}^n in which the matrix of $R^T R$ is as diagonal as possible. The problem is then solved in this basis

by using the corresponding ‘reduced form’ of R : \bar{R} ; the integer-valued solution \bar{v} thus obtained is then expressed in the original basis: $\bar{v} \mapsto \hat{v}$.

The first step corresponds to a decorrelation process. The decorrelation methods to be implemented must somehow refer to the principles of the LLL algorithm (an algorithm devised by Lenstra, Lenstra and Lovász in 1982). In the case of GNSS networks, as R may be of large size, particular implementations of this algorithm are to be devised.

For the particular methods presented in this section, the entries of v are lumped together in \mathbf{m} receiver blocks v_i :

$$v = \begin{bmatrix} v_1 \\ \vdots \\ v_i \\ \vdots \\ v_m \end{bmatrix} \quad (\mathbf{m} \leq m) \quad (88)$$

The receiver blocks are of the form

$$v_i := \begin{bmatrix} [N^{[\text{cd}]}]_{i,\nu_1} \\ [N^{[\text{cd}]}]_{i,\nu_2} \\ [N^{[\text{cd}]}]_{i,\nu_3} \end{bmatrix} \quad (89)$$

For example, at epoch t_k of Fig. 8, we have

$$[N^{[\text{cd}]}]_{i,\nu} := \left[\cdots \quad N_{\nu}^{[\text{cd}]}(i+1, j) \quad \cdots \right]^T \quad (90)$$

where j spans the subset of \mathcal{L}_{i+1} defined by the loop-closure points of line $i+1$; see Eq. (51). We then have $\mathbf{m} = m - 1 = 2$.

The number of entries of v_i is denoted by \mathbf{n}_i . For example, with three carrier waves on a full network (see Fig. 2), we have $\mathbf{n}_i = 3(n-1)$ for $i = 1, \dots, \mathbf{m}$ with $\mathbf{m} = m - 1$. Note that

$$\mathbf{n} = \sum_{i=1}^{\mathbf{m}} \mathbf{n}_i \quad (91)$$

The structure of R induced by that of v includes \mathbf{m} vertical bands of the form

$$B_1 := \begin{bmatrix} T_1 \\ 0 \\ \vdots \end{bmatrix} \quad B_i := \begin{bmatrix} S_i \\ T_i \\ 0 \\ \vdots \end{bmatrix} \quad (\text{for } i > 1) \quad (92)$$

Here, T_i is an upper-triangular matrix with \mathbf{n}_i positive diagonal elements; S_i is a rectangular matrix with \mathbf{p}_i lines and \mathbf{n}_i columns:

$$\mathbf{p}_i = \sum_{\iota=1}^{i-1} \mathbf{n}_\iota \quad (i > 1) \quad (93)$$

As specified in Sect. 8.1, the search for a reduced basis can be initialized via some inter-frequency decorrelation process. For each loop-closure point (i, j) of \mathcal{L}_i , the variables $N_{\nu_1}^{[\text{cd}]}(i, j)$, $N_{\nu_2}^{[\text{cd}]}(i, j)$, $N_{\nu_3}^{[\text{cd}]}(i, j)$ can thus be decorrelated. It is important to note that this process performs an operation basically similar to that of the widelane and extra-widelane techniques (see e.g., Feng and Li 2008, Teunissen 1997). This pointed out, by proceeding in this way for each T_i , one benefits from the correlation information concerning these variables at the current epoch.

It is however preferable to perform, directly, what we call ‘LLL band decorrelation’ (Sect. 8.2). This pointed out, once this weakened implementation of the LLL algorithm has been performed, the size of the search ellipsoid must generally be reduced. This is done via the ‘blockwise-bootstrapping’ method described in Sect. 8.3. The integer-ambiguity solution can then be obtained and validated via standard integer-programming techniques (Agrell et al. 2002).

8.1 Inter-frequency decorrelation

With regard to the frequency-block structure of v_i (see Eqs. (89) and (90)), the matrix elements of T_i relative to the same loop-closure point are then distributed as follows:

$$\begin{array}{cccc} \mathbf{t}_{\nu_1, \nu_1} & \cdots & \mathbf{t}_{\nu_1, \nu_2} & \cdots & \mathbf{t}_{\nu_1, \nu_3} \\ & \ddots & \vdots & & \vdots \\ & & \mathbf{t}_{\nu_2, \nu_2} & \cdots & \mathbf{t}_{\nu_2, \nu_3} \\ & & & \ddots & \vdots \\ & & & & \mathbf{t}_{\nu_3, \nu_3} \end{array} \quad (94)$$

By performing appropriate operations on R (see, e.g., Luk and Tracy 2008), the following conditions can easily be imposed:

$$\begin{array}{ll} \bar{\mathbf{t}}_{\nu_1, \nu_1} > 2|\bar{\mathbf{t}}_{\nu_1, \nu_2}| & \bar{\mathbf{t}}_{\nu_1, \nu_1} > 2|\bar{\mathbf{t}}_{\nu_1, \nu_3}| \\ \bar{\mathbf{t}}_{\nu_2, \nu_2} > 2|\bar{\mathbf{t}}_{\nu_2, \nu_3}| & \end{array} \quad (95)$$

Note that in this process, the diagonal elements are not modified: $\bar{\mathbf{t}}_{\nu, \nu} = \mathbf{t}_{\nu, \nu}$. For each receiver block (of index i), these operations are performed for each set of the three entries to be considered. The upper-triangular matrix \bar{R} thus obtained is equal to RZ where Z is a unimodular matrix. (By definition, a unimodular matrix is an integer matrix whose inverse is also an integer matrix.) We then have $Rv = \bar{R}\bar{v}$ where $\bar{v} = Z^{-1}v$. The entries of \bar{v} are the components of the integer-ambiguity vector in the reduced basis thus defined. This process also provides Z^{-1} .

8.2 LLL band decorrelation

The guiding idea of the decorrelation process presented in this section is to perform complete LLL decorrelations of the successive triangular blocks T_1, T_2, \dots, T_m .

The matrix elements of the blocks \bar{T}_i and \bar{S}_i of the decorrelated bands \bar{B}_i thus obtained are respectively denoted by $\bar{t}_i^{k,\ell}$ and $\bar{s}_i^{q,\ell}$ (see Eq. (92)). The diagonal elements of \bar{R} are then denoted by $\bar{r}^{q,q}$. For $i > 1$, we thus have $\bar{t}_i^{k,k} = \bar{r}^{\mathbf{p}_i+k, \mathbf{p}_i+k}$ for $1 \leq k \leq \mathbf{n}_i$.

The band decorrelation in question is a simple extension of the ‘new implementation’ of the LLL algorithm proposed by Luk and Tracy (2008). For clarity, we set

$$\mu_i^k := \left| \frac{\bar{t}_i^{k,k+1}}{\bar{t}_i^{k,k}} \right|^2 \quad (1 \leq k < \mathbf{n}_i) \quad (96)$$

The following conditions, with $1/4 < \varpi < 1$, can then be imposed:

For band $i = 1, \dots, \mathbf{m}$

$$\left\{ \begin{array}{l} \text{If } i > 1, \text{ then} \\ \text{for } q = \mathbf{p}_i, \mathbf{p}_i - 1, \dots, 1 \quad \bar{r}^{q,q} > 2|\bar{s}_i^{q,1}| \end{array} \right.$$

For column $\ell = 2, \dots, \mathbf{n}_i$ of \bar{B}_i

$$\left\{ \begin{array}{l} \mu_i^{\ell-1} < 1/4; \\ [\bar{t}_i^{\ell,\ell}]^2 \geq (\varpi - \mu_i^{\ell-1}) [\bar{t}_i^{\ell-1,\ell-1}]^2 \end{array} \right.$$

$$\text{If } \ell > 2, \text{ then} \\ \text{for } k = \ell - 2, \ell - 3, \dots, 1 \quad \bar{t}_i^{k,k} > 2|\bar{t}_i^{k,\ell}|$$

$$\text{For } q = \mathbf{p}_i, \mathbf{p}_i - 1, \dots, 1 \quad \bar{r}^{q,q} > 2|\bar{s}_i^{q,\ell}|$$

}

In fact, for optimal decorrelation, ϖ is set equal to 0.999.

This procedure provides \bar{R} as the product QRZ , in which Q is an orthogonal matrix, and Z is a unimodular matrix. We then have

$$\|Rv\|_{\mathbb{R}^n}^2 = \|\bar{R}\bar{v}\|_{\mathbb{R}^n}^2 \quad (97)$$

where

$$\bar{v} := Z^{-1}v \quad (98)$$

The entries of \bar{v} are the components of the ambiguity vector in the reduced basis thus defined; Z^{-1} is progressively built through the process (together with Z). It follows that

$$\|R(v - \hat{v})\|_{\mathbb{R}^n}^2 = \|\bar{R}(\bar{v} - \bar{\hat{v}})\|_{\mathbb{R}^n}^2 \quad (99)$$

where

$$\bar{\hat{v}} := Z^{-1}\hat{v} \quad (100)$$

In the absence of any prior information, the search ellipsoid (the ellipsoid in which the solution is to be searched) is then defined by the relation

$$E_0 := \{\bar{v} : \|\bar{R}(\bar{v} - \bar{\hat{v}})\|_{\mathbb{R}^n}^2 \leq \epsilon_0\} \quad (101)$$

where

$$\epsilon_0 := \|\bar{R}(\bar{v}^{(0)} - \bar{\hat{v}})\|_{\mathbb{R}^n}^2 \quad \text{with } \bar{v}^{(0)} := \lceil \bar{\hat{v}} \rceil \quad (102)$$

Here, $\lceil \bar{\hat{v}} \rceil$ denotes the column matrix whose entries are the nearest integers to the corresponding entries of $\bar{\hat{v}}$. Before solving the nearest-lattice point problem in the reduced basis, one may be led to reduce the size of the search ellipsoid. The blockwise-bootstrapping technique described below can then be implemented.

8.3 Blockwise bootstrapping

The procedure presented in this section provides a finite sequence of integer-ambiguity vectors $\bar{v}^{(q)}$ such that

$$\epsilon_{q+1} < \epsilon_q \quad \text{where } \epsilon_q := \|\bar{R}(\bar{v}^{(q)} - \bar{\hat{v}})\|_{\mathbb{R}^n}^2 \quad (103)$$

The ellipsoids

$$E_q := \{\bar{v} : \|\bar{R}(\bar{v} - \bar{\hat{v}})\|_{\mathbb{R}^n}^2 \leq \epsilon_q\} \quad (104)$$

therefore satisfy the property $E_{q_f} \subset \dots \subset E_1 \subset E_0$ where q_f is some finite integer.

To define this procedure, let us consider the quadratic functional (see Eqs. (99), (92) and (88))

$$\begin{aligned} \epsilon(\bar{v}_1, \dots, \bar{v}_i, \dots, \bar{v}_m) &:= \|\bar{R}(\bar{v} - \bar{\hat{v}})\|_{\mathbb{R}^n}^2 \\ &= \left\| \sum_{i=1}^m \bar{B}_i(\bar{v}_i - \bar{\hat{v}}_i) \right\|_{\mathbb{R}^n}^2 \end{aligned} \quad (105)$$

From Eq. (102), we have $\epsilon(\bar{v}_1^{(0)}, \dots, \bar{v}_i^{(0)}, \dots, \bar{v}_m^{(0)}) = \epsilon_0$. The principle of this procedure is then the following. For example, we first

$$\text{minimize } \epsilon(\bar{v}_1^{(0)}, \dots, \bar{v}_{m-1}^{(0)}, \bar{v}_m) \text{ in } \bar{v}_m \quad (106)$$

The minimum is attained for some $\bar{v}_m^{(1)} \in \mathbb{Z}^{\mathbf{n}_m}$. (How to do that is specified at the end of this section.) We then

$$\text{minimize } \epsilon(\bar{v}_1^{(0)}, \dots, \bar{v}_{m-2}^{(0)}, \bar{v}_{m-1}, \bar{v}_m^{(1)}) \text{ in } \bar{v}_{m-1} \quad (107)$$

The minimum is attained for some $\bar{v}_{m-1}^{(1)} \in \mathbb{Z}^{\mathbf{n}_{m-1}}$. We proceed like that until the first block variable included. We have thus found an ‘integer-ambiguity point’

$$\bar{v}^{(1)} := (\bar{v}_1^{(1)}, \dots, \bar{v}_i^{(1)}, \dots, \bar{v}_m^{(1)})$$

for which $\epsilon = \epsilon_1$ with $\epsilon_1 \leq \epsilon_0$. If $\epsilon_1 = \epsilon_0$, the process is interrupted. Otherwise, we then

$$\text{minimize } \epsilon(\bar{v}_1^{(1)}, \dots, \bar{v}_{m-1}^{(1)}, \bar{v}_m) \text{ in } \bar{v}_m \quad (108)$$

and so on until the process is interrupted.

We now specify how to perform the internal minimizations of type (106), (107) and (108). For example, with

regard to problem (106), we have, from Eq. (105),

$$\begin{aligned} & \epsilon(\bar{v}_1^{(0)}, \dots, \bar{v}_{m-1}^{(0)}, \bar{v}_m) \\ &= \left\| \bar{B}_m(\bar{v}_m - \bar{v}_m) + \sum_{i=1}^{m-1} \bar{B}_i(\bar{v}_i^{(0)} - \bar{v}_i) \right\|_{\mathbb{R}^n}^2 \\ &= \left\| \bar{B}_m \bar{v}_m - \left[\bar{B}_m \bar{v}_m - \sum_{i=1}^{m-1} \bar{B}_i(\bar{v}_i^{(0)} - \bar{v}_i) \right] \right\|_{\mathbb{R}^n}^2 \end{aligned}$$

The problem of minimizing this quantity in \bar{v}_m is solved in two steps: we first find its float solution by QR factorization; see Sect. 7.1. In the current reduced basis, we then solve the remaining problem of type (79); see Agrell et al. (2002). Note that for each i , the action of the operator Q_i^T involved in the first corresponding QR operation is to be stored in memory (see Sect. 7.1).

9 DIA methods

To prevent that biases on the undifferential data propagate undetected into the ambiguity solution, particular methods have been developed. The biases are first ‘Detected,’ then ‘Identified,’ and finally the results are ‘Adapted’ consequently (e.g., Teunissen 1990, Hewitson et al. 2004, Lannes and Gratton 2008). The identification principle of the DIA method presented in this section is ‘local:’ the biases are identified epoch by epoch. The corresponding analysis is based on the results provided by the QR process at the current epoch. When the ambiguities are not fixed, the adaptation principle is global: the local variables, the current biases, the current float ambiguities and the current QR triangular structure (sketched in Fig. 7) are updated in the global frame of the RLS process, without any approximation.

9.1 Quality control

Let us set (see Eqs. (45) and (46))

$$\hat{\theta}_{\psi,\kappa} := \tilde{\Psi}_{\psi,\kappa} - (\mathcal{A}_{\psi,\kappa} \hat{u}_{\kappa} + \mathcal{B}_{\psi,\kappa} \hat{v})$$

and

$$\hat{\varphi}_{\psi,\kappa}^{\circ} := \operatorname{argmin}_{\varphi \in F} \|\hat{\theta}_{\psi,\kappa} - \varphi\|_{\psi,\kappa}^2$$

When the model defined by Eqs (43) and (66) holds, on each point of G , $|\hat{\theta}_{\psi,\kappa} - \hat{\varphi}_{\psi,\kappa}^{\circ}|$ is then less than a few $\sigma_{\psi} \sqrt{\eta}$, say less than $\chi_0 \sigma_{\psi} \sqrt{\eta}$ where χ_0 is of the order of 3 for example; for further details on the choice of χ_0 , see Sect. 9.5. But, from Eqs. (48) and (47),

$$\hat{\theta}_{\psi,\kappa} - \hat{\varphi}_{\psi,\kappa}^{\circ} = \mathcal{P}'_{\psi,k} \hat{\theta}_{\psi,\kappa}$$

As a result (see Eq. (58)), the absolute value of

$$\mathcal{R}_{\psi,k} \hat{\theta}_{\psi,\kappa} \equiv U_{\psi,k} \mathcal{P}'_{\psi,k} \hat{\theta}_{\psi,\kappa}$$

is then less than χ_0 on G ; see Eq. (67). Taking account of Eqs. (61) and (62), we are thus led to concentrate on the quantity

$$w_{\psi,k} := \mathcal{R}_{\psi,k} \hat{\theta}_{\psi,\kappa} = \tilde{\Psi}_{\psi,\kappa}^{\mathbf{r}} - (\mathcal{A}_{\psi,\kappa}^{\mathbf{r}} \hat{u}_{\kappa} + \mathcal{B}_{\psi,\kappa}^{\mathbf{r}} \hat{v}) \quad (109)$$

If for $\psi = (\phi; \nu)$, $(p; \nu)$ and for each ν , $|w_{\psi,k}|$ is less than χ_0 all over G , we therefore consider that the model can be accepted as it is. Note that by construction, $w_{\psi,k}$ is the ψ -component of the local residual

$$\begin{aligned} w_k &:= b_k - (A_k \hat{u}_k + B_k \hat{v}) \\ &= H_k b_k \end{aligned} \quad (110)$$

where H_k is a linear operator (see Sect. 7).

For clarity, we now omit the time subscript k . In this context, to control the validity of the model, we consider the quantity

$$|w|_{\max} := \max_{\nu} \max_{\psi=(\phi;\nu)} \max_{(i,j) \in G} |w_{\psi}(i,j)| \quad (111)$$

If $|w|_{\max}$ is larger than χ_0 , the model is to be refined. For some ψ 's and some (i,j) 's to be identified, we then search to estimate additive biases $\beta_{\psi}(i,j)$. More precisely, the algebraic definition of these biases is such that the original data $\Psi_{\psi}(i,j)$ should then be corrected as follows:

$$\Psi_{\psi}(i,j) \stackrel{\text{set}}{=} \Psi_{\psi}(i,j) - \beta_{\psi}(i,j) \quad (112)$$

According to Eqs. (49) and (45), $\mathfrak{F}^{\mathbf{r}}$ and thereby w_{ψ} are invariant under any variation of Ψ_{ψ} in the vertex-delay space F ; see Fig. 4. As a result, to handle the identification problem in question, a preliminary notion is to be introduced: the notion of ‘identifiable bias.’

9.2 Identifiable biases

A unity bias on some receiver-satellite signal $\psi(i,j)$ is defined by the characteristic function $c_{i,j}$ of edge (r_i, s_j) :

$$c_{i,j}(i',j') := \begin{cases} 1 & \text{if } (i',j') = (i,j) \\ 0 & \text{otherwise} \end{cases} \quad (113)$$

Let us now consider two ‘signed unity biases’ on ψ having the same closure delays. As these biases are equal up to a vector of F , their reduced forms are identical. As a result, they cannot be distinguished. The following analysis clarifies this point explicitly.

The values of $c_{i,j}^{[\text{cd}]}$ on the loop-closure points of G form a vector $\mathbf{c}_{i,j}^{[\text{cd}]}$ including n_c components. As the closure delays are algebraic sums of SD's (see Remark 4.4), these components are equal to ± 1 or zero. The simplest way to determine them is to use the recursive differential process defined in Sect. 4.2. For our present purposes, we say that the one-dimensional subspace generated by $\mathbf{c}_{i,j}^{[\text{cd}]}$

defines a ‘bias direction’ \mathbf{d} in $E^{[\text{cd}]}$. As two distinct edges may define the same bias direction, the number of identifiable biases is less than or equal to the number of edges: $n_{\text{d}} \leq n_{\text{e}}$. For example, in the case of the graph defined in Fig. 3, n_{d} is equal to 6; the vectors \mathbf{d}_{ℓ} are then the following:

$$\begin{array}{cccccc} \mathbf{d}_1 & \mathbf{d}_2 & \mathbf{d}_3 & \mathbf{d}_4 & \mathbf{d}_5 & \mathbf{d}_6 \\ \hline 1 & 0 & 0 & 1 & -1 & 0 \\ 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 & -1 & 1 \end{array}$$

They are obtained in this order by spanning, first the n_{c} loop-closure points of G , and then the points of G_{st} . By construction, we thus have (see Eq. (32))

$$n_{\text{c}} \leq n_{\text{d}} \leq n_{\text{e}} \quad (114)$$

Denoting by G_{ℓ} the set of points of G whose bias direction is equal to \mathbf{d}_{ℓ} , we then say that $\bigcup_{\ell=1}^{n_{\text{d}}} G_{\ell}$ is the ‘bias partition’ of G . For example, in the special case of Fig. 3, we have

$$\begin{aligned} G_1 &= \{(2, 4)\} & G_2 &= \{(3, 3); (1, 3)\} & G_3 &= \{(3, 4)\} \\ G_4 &= \{(1, 1); (2, 1)\} & G_5 &= \{(1, 4)\} & G_6 &= \{(2, 2); (3, 2)\} \end{aligned}$$

The first element of G_{ℓ} is denoted by e_{ℓ} . Here for example, $e_2 = (3, 3)$; e stands for edge. Whenever G_{ℓ} includes two grid points, the latter are of the form (i, j) and (i', j) . Furthermore, we then have $\mathbf{c}_{i,j}^{[\text{cd}]} = \mathbf{d}_{\ell}$ and $\mathbf{c}_{i',j}^{[\text{cd}]} = -\mathbf{d}_{\ell}$. An identifiable bias is thus associated either with a receiver-satellite signal, or with a single difference. When $m = 2$, each bias is associated with a single difference. The number of identifiable biases is then equal to n : $n_{\text{d}} = n$; see Lannes and Gratton (2008). Conversely, in the case where the GNSS graphs are full or almost full with $m > 2$, n_{d} proves to be equal to n_{e} .

9.3 Identification principle

When the model is to be refined, we search to identify additive biases of the form $\beta_{\ell_{\psi}}$ (see Sect. 9.1); here, ℓ_{ψ} characterizes the corresponding ‘outlier:’ an outlier with direction \mathbf{d}_{ℓ} on the data vector ψ . The outliers ℓ_{ψ} form a set to be identified: the ‘outlier set’ \mathcal{O} .

According to Eqs. (110), (65) and (62), the variation of w induced by the unity bias $c_{e_{\ell}}$ on ψ is characterized by the quantity

$$f_{\ell_{\psi}} := H \begin{bmatrix} \vdots \\ 0 \\ [\mathcal{R}_{\psi} c_{e_{\ell}}] \\ 0 \\ \vdots \end{bmatrix} \quad (115)$$

As a result, the variation of w induced by the global bias

$$z := \sum_{\ell_{\psi} \in \mathcal{O}} \beta_{\ell_{\psi}} \begin{bmatrix} \vdots \\ 0 \\ [c_{e_{\ell}}] \\ 0 \\ \vdots \end{bmatrix} \quad (116)$$

is characterized by the vector

$$Mz := \sum_{\ell_{\psi} \in \mathcal{O}} \beta_{\ell_{\psi}} f_{\ell_{\psi}} \quad (117)$$

More precisely, from Eq. (112), w should then be corrected as follows: $w \stackrel{\text{set}}{=} w - Mz$. The problem is therefore to solve, in the LS sense, the equation $w - Mz \stackrel{!}{=} 0$, in which the column vectors of M , the $f_{\ell_{\psi}}$ ’s, are to be thoroughly selected. As clarified in Sect. 9.5, this operation is performed via a particular Gram-Schmidt orthogonalization process which is interrupted as soon as the corrected data are consistent with the model.

9.4 Global adaptation

Once the outlier set \mathcal{O} has been identified, the model is to be updated consequently: A_k is completed by adding the columns associated with the corresponding bias variables $\beta_{\ell_{\psi}}$. These column matrices have the following block form (see Eqs. (112), (65) and (62)):

$$\begin{bmatrix} \vdots \\ 0 \\ [\mathcal{R}_{\psi} c_{e_{\ell}}] \\ 0 \\ \vdots \end{bmatrix} \quad (118)$$

The global QR recursive process is then updated accordingly. The local variable, the biases and the float ambiguities are thus refined, as well as R_k and d_k in particular (see Fig. 6). When the QR process is initialized, or when the ambiguities are fixed, the biases provided by the adaptation process coincide with those provided by the identification procedure (see Sect. 9.3 and step 2.5 in Sect. 9.5). The LS problem to be solved, which is then the same, is simply handled in a different manner.

9.5 Implementation

In the procedure described in this section, the outliers ℓ_{ψ} are identified progressively; see the flow diagram shown in Fig. 10. At the beginning of this procedure, \mathcal{O} is therefore empty. For each ψ , we then introduce the set

$$\mathcal{K}_{\psi} := \{\ell : 1 \leq \ell \leq n_{\text{d}}\} \quad (119)$$

Setting (see Sect. 9.2)

$$C_{\psi,\ell} := \max_{(i,j) \in G_\ell} |w_\psi(i,j)| \quad (120)$$

we then consider the quantity

$$C_{\max} := \max_{\nu} \max_{\psi=(p;\nu)} \max_{\ell \in \mathcal{K}_\psi} C_{\psi,\ell} \quad (121)$$

At the beginning of the procedure, we therefore have $C_{\max} = |w|_{\max}$; see Eq. (111).

Given some probability of false alarm θ_0 , the threshold parameter χ_0 may be defined as the upper $\theta_0/2$ probability point of the central normal distribution:

$$\chi_0 := N_{\theta_0/2}(0,1)$$

For example, when θ_0 is equal to 0.001, χ_0 is of the order of 3.

This threshold parameter may also be defined, heuristically, as a given multiple of the mean value of $|w_\psi(i,j)|$ on G for all ψ .

1. Entrance test on C_{\max}

When C_{\max} is smaller than χ_0 , the model is accepted as it is: no outlier is to be searched; one then goes to step 4. Conversely, if C_{\max} is very large compared to χ_0 (say larger than 1000 for example), the QR process is reinitialized (see Sect. 7). In the other cases, the DIA procedure is initialized by setting $\tau = 1$ and $\Pi = \emptyset$; τ is a recursive index; the meaning of the auxiliary set Π is defined in step 2.2 as soon as it begins to be built. At this stage, the ‘local redundancy’ of the problem, L_r , has a given value.

2. Recursive identification of the outliers

2.1. Current set of potential outliers

Given some nonnegative constant $\kappa \leq 1$, form the current set of potential outliers

$$\Pi_\tau := \bigcup_{\nu} \bigcup_{\psi=(\phi;\nu)}^{\psi=(p;\nu)} \{l_\psi : l_\psi \in \mathcal{K}_\psi, C_{\psi,\ell} \geq \kappa C_{\max}\}$$

2.2. For each potential outlier $l_\psi \in \Pi_\tau$

Perform the following successive operations:

a) When $l_\psi \notin \Pi$, compute f_{l_ψ} ; to do that, see the context of Eqs. (115), (110) and Sect. 7.3. Then, set

$$g_{l_\psi} = f_{l_\psi} \quad \Pi \stackrel{\text{set}}{=} \begin{cases} \{l_\psi\} & \text{if } \Pi = \emptyset \\ \Pi \cup \{l_\psi\} & \text{otherwise} \end{cases}$$

By construction, Π is the set of potential outliers l_ψ for which f_{l_ψ} has already been computed.

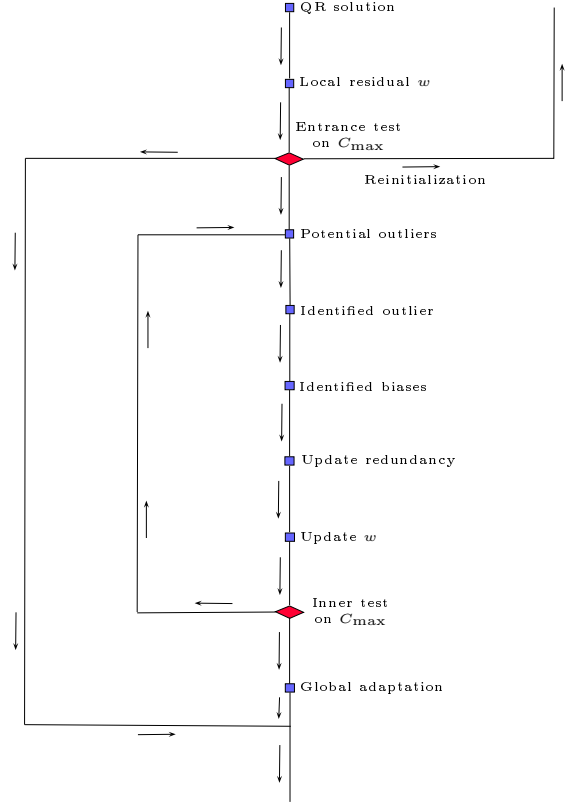


Figure 10: *Flow diagram of the DIA procedure.* This procedure is based on an examination of the local residual w . This residual, with components $w_\psi(i,j)$, is the observational residual simply divided by the standard deviation $\sigma_\psi \sqrt{\eta(i,j)}$ of the original data; see the introduction of Eq. (109). At each step of the identification process, the updated values of w are analyzed on the grounds of Eqs. (121) and (120); see steps 1, 2.7 and 2.8. This allows the potential outliers to be selected. The outliers can thus be identified, in a recursive manner, via a particular orthogonalization Gram-Schmidt process. This QR Gram-Schmidt process also provides the identifiable biases (see Sect. 9.2), and thereby the cycle slips if any. When the ambiguity are not fixed, these biases are slightly refined through the global adaptation process described in Sect. 9.4.

b) If $\tau = 1$ go to step 2.2c. Otherwise, at this level, $\{g_q^\circ\}_{q < \tau}$ is an orthonormal set. (This set is built, progressively, via step 2.3.) Then, for each integer $q < \tau$, consider the inner product defined as follows:

$$\varsigma_{q,l_\psi} := (g_q^\circ \cdot g_{l_\psi}) \equiv [g_q^\circ]^T [g_{l_\psi}]$$

If ς_{q,l_ψ} has not been computed yet, compute it, store it in memory, and perform the Gram-Schmidt orthogonalization operation

$$g_{l_\psi} \stackrel{\text{set}}{=} g_{l_\psi} - \varsigma_{q,l_\psi} g_q^\circ$$

By construction,

$$\varsigma_{q,l_\psi} = (g_q^\circ \cdot f_{l_\psi})$$

At the end of all these operations, g_{l_ψ} is orthogonal to g_q° for any $q < \tau$.

c) Consider the projection of w on the one-dimensional space generated by g_{ℓ_ψ} , i.e.,

$$(h_{\ell_\psi} \cdot w)h_{\ell_\psi} \quad h_{\ell_\psi} := g_{\ell_\psi} / \|g_{\ell_\psi}\|$$

where $\|g_{\ell_\psi}\|^2 \equiv [g_{\ell_\psi}]^T [g_{\ell_\psi}]$. The norm of this projection is equal to $|(h_{\ell_\psi} \cdot w)|$, the absolute value of the quantity

$$\gamma_{\ell_\psi} := (g_{\ell_\psi} \cdot w) / \varrho_{\ell_\psi} \quad \varrho_{\ell_\psi} := \|g_{\ell_\psi}\|$$

2.3. Identified outlier

The identified outlier $\ell_{\psi^*}^*$ is defined as the dominant potential outlier, i.e., the potential outlier for which $|\gamma_{\ell_\psi}|$ is maximal:

$$\ell_{\psi^*}^* := \arg \max_{\ell_\psi \in \Pi_\tau} |\gamma_{\ell_\psi}|$$

We then discard $\ell_{\psi^*}^*$ from \mathcal{K}_{ψ^*} : $\mathcal{K}_{\psi^*} \stackrel{\text{set}}{=} \mathcal{K}_{\psi^*} - \{\ell_{\psi^*}^*\}$. We then set

$$\circ_\tau := \ell_{\psi^*}^* \quad \mathcal{O} \stackrel{\text{set}}{=} \begin{cases} \{\circ_\tau\} & \text{if } \tau = 1 \\ \mathcal{O} \cup \{\circ_\tau\} & \text{if } \tau > 1 \end{cases}$$

$$\gamma_\tau^\circ := \gamma_{\circ_\tau} \quad g_\tau^\circ := g_{\circ_\tau} / \varrho_{\circ_\tau}$$

Here, \circ stands for outlier. At this level, \mathcal{O} is the current set of identified outliers:

$$\mathcal{O} = \{\circ_q\}_{q=1}^\tau$$

By construction, $\{g_q^\circ\}_{q=1}^\tau$ is an orthonormal basis of the current range of M ; $\sum_{q=1}^\tau \gamma_q^\circ g_q^\circ$ is the projection of w on this space. With regard to Eq. (117), we then set

$$\beta_\tau^\circ := \beta_{\circ_\tau} \quad f_\tau^\circ := f_{\circ_\tau}$$

2.4. Components of g_τ° in the basis of the f_q° 's

These components are denoted by $u_{q,\tau}$:

$$g_\tau^\circ = \sum_{q=1}^\tau u_{q,\tau} f_q^\circ$$

They are computed via the following QR Gram-Schmidt formulas (see, e.g., Björck 1996):

$$u_{q,\tau} = \begin{cases} -\frac{1}{\varrho_{\circ_\tau}} \sum_{q \leq q' < \tau} u_{q,q'} s_{q',\circ_\tau} & \text{if } q < \tau \\ \frac{1}{\varrho_{\circ_\tau}} & \text{if } q = \tau \end{cases}$$

for $1 \leq q \leq \tau$. The $u_{q,\tau}$'s are the entries of the τ^{th} column of an upper triangular matrix U .

2.5. Identified biases

According to Eq. (117), the biases β_q° are the components of $\sum_{q=1}^\tau \gamma_q^\circ g_q^\circ$ in the basis of the f_q° 's:

$$\sum_{q=1}^\tau \gamma_q^\circ g_q^\circ = \sum_{q=1}^\tau \beta_q^\circ f_q^\circ$$

Denoting by $[\gamma^\circ]$ the column matrix with entries γ_q° (from $q = 1$ to τ), and likewise for $[\beta^\circ]$, we have $[\beta^\circ] = U[\gamma^\circ]$. The identified biases are therefore to be updated as follows:

$$\beta_q^\circ \stackrel{\text{set}}{=} \begin{cases} \beta_q^\circ + u_{q,\tau} \gamma_\tau^\circ & \text{if } q < \tau \\ u_{\tau,\tau} \gamma_\tau^\circ & \text{if } q = \tau \end{cases} \quad (\text{for } 1 \leq q \leq \tau)$$

2.6. Update the local redundancy

$$Lr \stackrel{\text{set}}{=} Lr - 1$$

If $Lr = 0$ go to step 3.

2.7. Update w

$$w \stackrel{\text{set}}{=} w - \gamma_\tau^\circ g_\tau^\circ$$

2.8. Update C_{\max}

$$C_{\max} := \max_{\nu} \max_{\substack{\psi=(p;\nu) \\ \psi=(\phi;\nu)}} \max_{\ell \in \mathcal{K}_\psi} C_{\psi,\ell}$$

2.9. Inner test on C_{\max}

If $C_{\max} > \chi_0$, update the recursive index: $\tau \stackrel{\text{set}}{=} \tau + 1$. Then, go to step 2.1.

3. Global adaptation

Update the global QR recursive process by taking account of the identified bias variables (see Sect. 9.4).

4. End

10 Concluding comments

The GNSS centralized approach presented in Lannes and Gratton (2008) was restricted to the case of RTK observations with a single baseline of local scale. That approach was validated by processing real GPS data in dual-frequency mode. The present paper was devoted to the extension of that contribution to the general case of multiple-baseline networks of any scale. (The extended satellite-clock biases are not estimated.)

To introduce the reader to the related concepts, we first examined the special case where the GNSS graph is full: all the receiver-satellite signals of the GNSS network are then available; see Sect. 3.1. The carrier-phase integer ambiguity vector can then be decomposed into three integer ambiguity components: the receiver, the satellite and the DD ambiguity vectors; see Eq. (23).

The heart of the paper concerned the extension of this property to the general case where the GNSS graph is not full. This was done in Sect. 4 with the aid of elementary notions of algebraic graph theory. From a technical point of view, the algebraic operations performed on the edges of the GNSS graph are to be followed on the corresponding GNSS grid; see Fig. 1. In this framework, as clarified in Sect. 4.2, the notion of closure delay generalizes that of double difference. In practice, according to their definition (Eq. (37)), the closure delays are computed via the recursive differential process defined in Sect. 4.2; see Example 4.3. This pointed out, a closure delay is an algebraic sum of single differences; see the example given in Remark 4.4. This shows that generalization does necessarily make the problems more complex. Instead, it provides a theoretical framework which leads to a better understanding of the matter, and thereby to more efficient techniques.

As specified in Sects. 5 and 6, the problem is stated and solved in terms of reduced quantities. The notion of reduction is closely related to that of centralization. This operation simply amounts to solving a linear system, the size of which is at most equal to the number of satellites other than the reference satellite; see Eqs. (73) and (72).

In the approach adopted in this paper, the float solution is refined, recursively, by using the QR method; see Sect. 7. The state transitions of the global variable, in particular those due to a change of the GNSS graph, have been examined in this framework. The example studied in Sect. 7.4.3 concerns complex circumstances. The notion of transition spanning tree is then essential for solving the problem in an elegant and efficient manner; see Fig. 8.

At each epoch, the QR method provides, in particular, the float ambiguity solution and the Cholesky factor of the inverse of its variance-covariance matrix. To solve the corresponding integer-ambiguity problem, this upper-triangular matrix is to be decorrelated. As the size of this matrix may be very large, a particular implementation of the LLL algorithm was proposed; see Sect. 8.2.

As shown in Sect. 9, the centralized mode is particularly well suited to the QR implementation of the DIA methods. On each edge of the GNSS graph, or equivalently, on each point of the corresponding grid, the observational residual is then simply divided by the standard deviation of the corresponding data; see the context of Eq. (109). The search for the potential outliers is then performed by simple inspection of the absolute value of these reduced quantities; see Eqs. (121), (120) and step 2.8 in Sect. 9.5. The statistical tests are thereby very simple; see steps 1 and 2.9 in Sect. 9.5. This pointed out, when some data are missing, the notion of identifiable bias is to be taken into account; see Sect. 9.2.

The operations involved in the selected QR implementation can be stored in memory very easily; see Sect. 7.3. As a result, the variational calculations involved in the

DIA methods can be performed in a very efficient manner; see step 2.2 in Sect. 9.5. Furthermore, the QR global adaptation step of the DIA method nicely completes the QR Gram-Schmidt step 2.4 of the local identification process described in Sect. 9.5. The identifiable biases, among which the cycles slips (if any), are thus estimated in two different ways; see Lannes and Gratton (2008).

References

- Agrell E., Eriksson T., Vardy A. and Zeger K. (2002) *Closest point search in lattices*. IEEE Trans. Inform. Theory. 48: 2201–2214.
- Bierman G.J. (1977) *Factorization methods for discrete sequential estimation*, Vol. 128 in Mathematics in science and engineering, Academic Press, Inc. New-York.
- Biggs N. (1996) *Algebraic graph theory*, 2nd edn Cambridge University Press, Cambridge.
- Björck A. (1996) *Numerical methods for least-squares problems*, SIAM.
- Chang X.-W. and Guo Y. (2005) *Huber's estimation in relative GPS positioning: computational aspects*. J. Geod. 79: 351–362.
- Feng Y. and Li B. (2008) *Three-carrier ambiguity resolution: generalized problems, models, methods and performance analysis using semi-generated triple frequency GPS data*. Proc. ION GNSS-2008. Savannah, Georgia USA: 2831–2840
- Golub G.H. and van Loan C.F. (1989) *Matrix computations*, second edition, The Johns Hopkins University Press, Baltimore, Maryland.
- Hewitson S., Lee H.K. and Wang J. (2004) *Localizability analysis for GPS/Galileo receiver autonomous integrity monitoring*. The Journal of Navigation, Royal Institute of Navigation 57: 245–259.
- Lannes A. (2005) *A global analysis of the phase calibration operation*. J. Opt. Soc. Am. A. 22: 697–707.
- Lannes A. (2008) *GNSS networks with missing data: identifiable biases and potential outliers*. Proc. ENC GNSS-2008. Toulouse, France.
- Lannes A. and Gratton S. (2008) *QR implementation of GNSS centralized approaches*. J. GPS. 7: 133–147.
- Liu X. (2002) *A comparison of stochastic models for GPS single differential kinematic positioning*. Proc. ION GPS-2002. Portland, Oregon USA: 1830–1841.
- Luk F.T. and Tracy D.M. (2008) *An improved LLL algorithm*. Linear algebra and its applications. 428: 441–452.

- Mercier F. and Laurichesse D. (2008) *Zero-difference ambiguity blocking. Properties of receiver-satellite biases*. Proc. ENC GNSS-2008. Toulouse, France.
- Shi P. H. and Han S. (1992) *Centralized undifferential method for GPS network adjustment*. Australian Journal of Geodesy, Photogrammetry and Surveying. 57: 89-100.
- Teunissen P.J.G. (1990) *An integrity and quality control procedure for use in multi sensor integration*. Proc. ION GPS-90. Colorado Springs, Colorado USA: 513-522
- Teunissen P.J.G. (1997) *On the GPS widelane and its decorrelating property*. J. Geod. 71: 577-587.
- Tiberius C.C.J.M. (1998) *Recursive data processing for kinematic GPS surveying*, Publications on Geodesy, New series: ISSN 0165 1706, Number 45, Netherlands Geodetic Commission, Delft.
-