

GNSS algebraic structures

A. Lannes · P. J. G. Teunissen

Received: 26 August 2010 / Accepted: 13 December 2010 / Published online: 5 January 2011
© The Author(s) 2011. This article is published with open access at Springerlink.com

Abstract The first objective of this paper is to show that some basic concepts used in global navigation satellite systems (GNSS) are similar to those introduced in Fourier synthesis for handling some phase calibration problems. In experimental astronomy, the latter are at the heart of what is called ‘phase closure imaging.’ In both cases, the analysis of the related structures appeals to the algebraic graph theory and the algebraic number theory. For example, the estimable functions of carrier-phase ambiguities, which were introduced in GNSS to correct some rank defects of the undifferenced equations, prove to be ‘closure-phase ambiguities:’ the so-called ‘closure-delay’ (CD) ambiguities. The notion of closure delay thus generalizes that of double difference (DD). The other estimable functional variables involved in the phase and code undifferenced equations are the receiver and satellite pseudo-clock biases. A related application, which corresponds to the second objective of this paper, concerns the definition of the clock information to be broadcasted to the network users for their precise point positioning (PPP). It is shown that this positioning can be achieved by simply having access to the satellite pseudo-clock biases. For simplicity, the study is restricted to relatively small networks. Concerning the phase for example, these biases then

include five components: a frequency-dependent satellite-clock error, a tropospheric satellite delay, an ionospheric satellite delay, an initial satellite phase, and an integer satellite ambiguity. The form of the PPP equations to be solved by the network user is then similar to that of the traditional PPP equations. As soon as the CD ambiguities are fixed and validated, an operation which can be performed in real time via appropriate decorrelation techniques, estimates of these float biases can be immediately obtained. No other ambiguity is to be fixed. The satellite pseudo-clock biases can thus be obtained in real time. This is not the case for the satellite-clock biases. The third objective of this paper is to make the link between the CD approach and the GNSS methods based on the notion of double difference. In particular, it is shown that the information provided by a maximum set of independent DDs may not reach that of a complete set of CDs. The corresponding defect is analyzed. One of the main results of the corresponding analysis concerns the DD–CD relationship. In particular, it is shown that the DD ambiguities, once they have been fixed and validated, can be used as input data in the ‘undifferenced CD equations.’ The corresponding algebraic operations are described. The satellite pseudo-clock biases can therefore be also obtained via particular methods in which the notion of double differencing is involved.

A. Lannes (✉)
CNRS, SUPELEC, Univ Paris-Sud, Supélec, 2 rue E. Belin,
57070 Metz, France
e-mail: andre.lannes@cerfacs.fr

P. J. G. Teunissen
Delft University of Technology, Kluyverweg 1, Delft,
The Netherlands

P. J. G. Teunissen
Curtin University of Technology, GPO Box U1987, Perth,
WA 6845, Australia
e-mail: p.teunissen@curtin.edu.au

Keywords GNSS networks · Phase calibration · Undifferenced and differenced methods · PPP · RTK

1 Introduction

The global positioning techniques are based on two types of data: the (carrier-)phase and code (or pseudo-range) observations. With n_p carrier waves, we first concentrate on the phase

equations. For each frequency ν , at each epoch k (where k stands for t_k), and for each receiver–satellite pair $(i, j) \equiv (r_i, s_j)$, these equations are of the form (see, e.g., [Teunissen and Kleusberg 1998](#))

$$\begin{aligned} \Phi_{\nu,k}(i, j) &= \rho_k(i, j) + T_k(i, j) - \kappa_\nu I_k(i, j) \\ &\quad + [\delta t_{\phi;\nu,k}^{(r)}(i) - \delta t_{\phi;\nu,k}^{(s)}(j)] \\ &\quad + \lambda_\nu[\varphi^{(r)}(i) - \varphi^{(s)}(j) + N_\nu(i, j)] \\ &\quad + \varepsilon_{\phi;\nu,k}(i, j) \end{aligned} \quad (1)$$

where

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

The expectation values of the noise terms $\varepsilon_{\phi;\nu,k}(i, j)$ are supposed to be zero. Here, the phase data $\Phi_{\nu,k}(i, j)$ are expressed in length units; $\rho_k(i, j)$ is the receiver–satellite range: the distance between satellite s_j (at the time $t_k - \Delta t_k$ where the signal is emitted) and receiver r_i (at the time t_k of its reception). The quantities $T_k(i, j)$ and $I_k(i, j)$ are the tropospheric and ionospheric delays, respectively. The λ_ν s denote the wavelengths of the carrier waves involved in the observational process. Note that $\kappa_{\nu_1} = 1$. The integers $N_\nu(i, j)$ are the integer carrier-phase ambiguities: $N_\nu(i, j) \in \mathbb{Z}$; $\varphi^{(r)}(i)$ and $\varphi^{(s)}(j)$ are the initial phases at receiver r_i and satellite s_j , respectively. These phases are expressed in cycles. The instrumental biases and the clock errors depending on r_i , ν and k are lumped together in the frequency-dependent receiver-clock errors $\delta t_{\phi;\nu,k}^{(r)}(i)$. Here, the latter are expressed in length units. Likewise, the instrumental biases and the clock errors depending on s_j , ν , and k are lumped together in the frequency-dependent satellite-clock errors $\delta t_{\phi;\nu,k}^{(s)}(j)$. In this paper, we consider the general case where we have no a priori information on the functions $\delta t_{\phi;\nu,k}^{(r)}$, $\delta t_{\phi;\nu,k}^{(s)}$, $\varphi^{(r)}$, and $\varphi^{(s)}$.

Just for the purpose of illustrating our approach, we now make some simplifying assumptions concerning the GNSS network. These assumptions do not reduce the general applicability, but for our present objectives simplify the derivations somewhat. The study is restricted to relatively small networks. More precisely, we assume that $T_k(i, j)$ and $I_k(i, j)$ do not depend on i :

$$T_k(i, j) = T_k(j), \quad I_k(i, j) = I_k(j) \quad (3)$$

Moreover, the positions of the receivers and satellites are assumed to be known. Equation (1) is then of the form

$$\begin{aligned} \Phi_{\nu,k}^*(i, j) &= [\delta t_{\phi;\nu,k}^{(r)}(i) - \delta t_{\phi;\nu,k}^{(s)}(j)] - \kappa_\nu I_k(j) \\ &\quad + \lambda_\nu[\varphi^{(r)}(i) - \varphi^{(s)}(j) + N_\nu(i, j)] \\ &\quad + \varepsilon_{\phi;\nu,k}(i, j) \end{aligned} \quad (4)$$

where

$$\Phi_{\nu,k}^*(i, j) = \Phi_{\nu,k}(i, j) - \rho_k(i, j) \quad (5)$$

and

$$\delta \tau_{\phi;\nu,k}^{(s)}(j) := \delta t_{\phi;\nu,k}^{(s)}(j) - T_k(j) \quad (6)$$

Concerning the code, we then have similarly

$$\begin{aligned} P_{\nu,k}^*(i, j) &= [\delta t_{p;\nu,k}^{(r)}(i) - \delta t_{p;\nu,k}^{(s)}(j)] + \kappa_\nu I_k(j) \\ &\quad + \varepsilon_{p;\nu,k}(i, j) \end{aligned} \quad (7)$$

where

$$P_{\nu,k}^*(i, j) = P_{\nu,k}(i, j) - \rho_k(i, j) \quad (8)$$

and

$$\delta \tau_{p;\nu,k}^{(s)}(j) := \delta t_{p;\nu,k}^{(s)}(j) - T_k(j) \quad (9)$$

The operator of the linear problem defined by the undifferenced equations (4) and (7) is not of full rank. The null space of this operator can be identified by showing which parameter changes leave the observations invariant. The corresponding rank deficiency can then be eliminated through an appropriate reduction and redefinition of the unknown parameters. The latter are the estimable functions of parameters of the minimum-constrained problem thus defined. The most general method associated with this analysis is the S -basis technique developed by [Baarda \(1973\)](#), [Teunissen \(1984\)](#), and [de Jonge \(1998\)](#).

The first objective of this paper is to make the link with a method developed in experimental astronomy for solving a similar problem: the phase calibration problem. This problem plays a key role when imaging brightness sources at high resolution via Fourier synthesis techniques; see [Remark 3.1](#). In fact, this calibration problem is at the heart of what is called ‘phase closure imaging’ (PCI) (see, e.g., [Lannes 2005](#)). To introduce the elements involved in this link, we first reduce the rank deficiency of Eqs. (4) and (7). This is done in [Sect. 2](#) by performing some elementary preliminary parametrization. Concerning the phase, we are then left with a key problem. The latter, which is presented in [Sect. 3](#), can be handled in two different ways. The first one derives from the approach developed in experimental astronomy (Approach 1). The GNSS version of the corresponding theoretical framework is introduced in [Sect. 4](#). The second approach corresponds to the implementation of the S -basis technique for correcting the rank defect of the key problem in question (Approach 2). The equivalence of these two approaches is established through [Sects. 5](#) and [6](#). In fact, as phase calibration problems are embedded in the processing of GNSS signals, the concepts introduced in these two approaches lead to a better understanding of these topics. For example, the PCI concept of ‘closure-phase ambiguity’ is equivalent to that of ‘estimable function of carrier-phase ambiguities.’ These estimable functions are therefore ‘closure-delay’ (CD) ambiguities. The notion of closure delay

thus generalizes that of double difference (DD). As specified in Sect. 7, the other estimable functions of parameters involved in the undifferenced equations are the receiver and satellite pseudo-clock biases. How to solve these equations in the least-square (LS) sense is summarized in Sect. 8.

A related application, which corresponds to the second objective of this paper, concerns the definition of the clock information to be broadcasted to the network users for their precise point positioning (PPP). The corresponding study is presented in Sect. 9. It is then shown that this precise positioning can be achieved by simply having access to the satellite pseudo-clock biases provided by the network. For example, concerning the phase, these pseudo-clock biases include five components: a frequency-dependent satellite-clock error, a tropospheric satellite delay, an ionospheric satellite delay, an initial satellite phase, and an integer satellite ambiguity. The form of the PPP equations to be solved by the user is then similar to that of the traditional PPP equations (for the corresponding standard approaches see, e.g., Zumberge et al. 1997; Ge et al. 2008; Bertiger et al. 2010). As soon as the CD ambiguities are fixed and validated (see Sect. 8), estimates of the satellite pseudo-clock biases can be immediately obtained. No other ambiguity is then to be fixed. These estimates can therefore be obtained in real time. Note that this is not the case for the satellite-clock biases (see, e.g., Bertiger et al. 2010; Dow et al. 2009).

The third objective of this paper is to make the link between the CD approach and the GNSS methods based on the notion of double difference. In particular, it is shown in Sect. 10 that the information provided by a maximum set of independent DDs may not reach that of a complete set of CDs. The corresponding defect is analyzed. One of the main results of the corresponding analysis concerns the DD–CD relationship. In particular, it is shown that the DD ambiguities, once they have been fixed and validated, can be used as input data in the ‘undifferenced CD equations.’ The corresponding algebraic operations are described. The satellite pseudo-clock biases can therefore be also obtained via particular methods in which the notion of double differencing is involved. Comments on the key points of our contribution are to be found in Sect. 11.

2 Preliminary parametrization

The rank deficiency of Eq. (4) can be reduced through the elementary parametrization

$$\Phi_{v,k}^*(i, j) = [\alpha_{\phi;v,k}^{[r]}(i) - \alpha_{\phi;v,k}^{[s]}(j)] + \lambda_v N_v(i, j) + \varepsilon_{\phi;v,k}(i, j) \quad (10)$$

in which

$$\alpha_{\phi;v,k}^{[r]}(i) := \alpha_{\phi;v,k}^{(r)}(i) - \alpha_{\phi;v,k}^{(s)}(1) \quad (11)$$

$$\alpha_{\phi;v,k}^{[s]}(j) := \alpha_{\phi;v,k}^{(s)}(j) - \alpha_{\phi;v,k}^{(s)}(1) \quad (12)$$

where

$$\alpha_{\phi;v,k}^{(r)}(i) := \delta t_{\phi;v,k}^{(r)}(i) + \lambda_v \varphi^{(r)}(i) \quad (13)$$

$$\alpha_{\phi;v,k}^{(s)}(j) := \delta \tau_{\phi;v,k}^{(s)}(j) + \kappa_v \mathbf{I}_k(j) + \lambda_v \varphi^{(s)}(j) \quad (14)$$

The code equation (7) is then parametrized similarly:

$$P_{v,k}^*(i, j) = [\alpha_{p;v,k}^{[r]}(i) - \alpha_{p;v,k}^{[s]}(j)] + \varepsilon_{p;v,k}(i, j) \quad (15)$$

Here,

$$\alpha_{p;v,k}^{[r]}(i) := \alpha_{p;v,k}^{(r)}(i) - \alpha_{p;v,k}^{(s)}(1) \quad (16)$$

$$\alpha_{p;v,k}^{[s]}(j) := \alpha_{p;v,k}^{(s)}(j) - \alpha_{p;v,k}^{(s)}(1) \quad (17)$$

where

$$\alpha_{p;v,k}^{(r)}(i) := \delta t_{p;v,k}^{(r)}(i) \quad (18)$$

$$\alpha_{p;v,k}^{(s)}(j) := \delta \tau_{p;v,k}^{(s)}(j) - \kappa_v \mathbf{I}_k(j) \quad (19)$$

In this parametrization, according to Eqs. (11), (12), (16) and (17), s_1 is the satellite that defines the pseudo-clock error references for the phase and the code. Indeed, we have introduced the constraints

$$\alpha_{\phi;v,k}^{[s]}(1) = 0, \quad \alpha_{p;v,k}^{[s]}(1) = 0 \quad (20)$$

Other constraints can equally well be imposed. As suggested by this preliminary parametrization, the pseudo-clock (functional) variables $\alpha_{\mu;v,k}^{[r]}$ and $\alpha_{\mu;v,k}^{[s]}$ (for $\mu = \phi$ or p) will play an important role.

3 The key problem

Concentrating on Eq. (10), let us set

$$\alpha^{[r]}(i) \equiv \alpha_{\phi;v,k}^{[r]}(i), \quad \alpha^{[s]}(j) \equiv \alpha_{\phi;v,k}^{[s]}(j) \\ \lambda \equiv \lambda_v, \quad N(i, j) \equiv N_v(i, j), \quad \varepsilon(i, j) \equiv \varepsilon_{\phi;v,k}(i, j)$$

and

$$\vartheta(i, j) \equiv \Phi_{v,k}^*(i, j)$$

Up to the error term $\varepsilon(i, j)$, Eq. (10) is then read as

$$\vartheta(i, j) = [\alpha^{[r]}(i) - \alpha^{[s]}(j)] + \gamma(i, j) \quad (21)$$

where $\gamma(i, j) = \lambda N(i, j)$.

The following preliminary problem can then be addressed: ϑ being known, to which extent is it possible to solve Eq. (21), with $\alpha^{[s]}(1) = 0$, $\alpha^{[r]}$, $\alpha^{[s]}$ and γ being real-valued functional variables. Clearly, $\alpha^{[r]}$ is defined by m real parameters to be determined: $\alpha^{[r]}(1), \dots, \alpha^{[r]}(m)$. Likewise, $\alpha^{[s]}$ is defined by $n - 1$ unknown parameters: $\alpha^{[s]}(2), \dots, \alpha^{[s]}(n)$. In the ‘float problem’ in question, γ is characterized by n_e

values to be somehow identified. In Approach 2, the guiding idea is then to correct for the rank defect of Eq. (21). As already pointed out, this approach is induced by the analysis developed by Baarda (1973) and Teunissen (1984) for correcting larger rank defects.

Without any reference to rank defects, a relation such as Eq. (21) can equally well be regarded as a particular decomposition of ϑ to be defined. The corresponding approach is referred to as Approach 1. It corresponds to the analysis presented in Lannes and Gratton (2009). As clarified via Sects. 4 and 5, a key notion is then introduced: the notion of closure delay (CD).

Remark 3.1 Approach 1 derives from a similar problem encountered in experimental astronomy: the problem of phase calibration. This operation plays a key role when imaging brightness sources at high resolution via Fourier synthesis techniques (see, e.g., Lannes 2005). This problem, which is at the heart of what is called ‘phase closure imaging’ (PCI), is to identify some phase biases $\delta(i)$. In the absence of noise, the phasor $\exp\{i\phi\}$ of the ‘experimental complex visibility function’ of the calibrator (a reference point-source brightness distribution) is then related to δ by a relationship of the form

$$\exp\{i\phi(i, j)\} = \exp\{i[\delta(i) - \delta(j)]\} \tag{22}$$

Here, the pairs (i, j) correspond to the baselines of the interferometric device; for further details (not related to the objectives of the present purposes), see Sect. 1 in Lannes (2005). The relationship between the phases ϕ and δ is therefore defined modulo 2π :

$$\phi(i, j) = [\delta(i) - \delta(j)] + 2\pi N(i, j) \tag{23}$$

Setting

$$\vartheta := \lambda \frac{\phi}{2\pi}, \quad \alpha := \lambda \frac{\delta}{2\pi} \tag{24}$$

we then have

$$\vartheta(i, j) = [\alpha(i) - \alpha(j)] + \gamma(i, j) \tag{25}$$

where $\gamma(i, j) = \lambda N(i, j)$.

Clearly, Eqs. (21) and (25) share a common feature. Phase calibration problems are therefore embedded in the processing of GNSS signals.

4 Theoretical framework

In Sect. 4.1, we first introduce the notions GNSS grid and GNSS graph. Section 4.2 is devoted to the concepts of spanning tree and loops. We then define related spaces of functions (Sect. 4.3).

4.1 GNSS grid and graph

The GNSS network includes m stations, and thereby m multi-frequency receivers r_i . The number of satellites s_j involved in the observational process over some time interval $[t_1, t_k] \equiv [1, k]$ is denoted by n . The ‘observational grid’ of the network is therefore a grid G_o including m lines, n columns, and mn points; see Fig. 1. For example, in the case of large networks, m and n are of the order of 100 and 32, respectively. A function such as $\Phi_{v,k}$ or ρ_k , with k in $[1, k]$, takes its values on some points (i, j) of G_o . These points form a subset of G_o denoted by G_k , the grid that characterizes the scenario at epoch k : the ‘GNSS grid.’ When no confusion may arise, subscript k is omitted: $G \equiv G_k$.

In the example presented in the upper part of Fig. 1, the points (i, j) of G are shown as black dots. As illustrated in the lower part of this figure, these points 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:

$$n_v = m + n \tag{26}$$

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

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

For simplicity, we now assume that \mathcal{G} is connected (see, e.g., Biggs 1996): given any two vertices of \mathcal{V} , there exists a path of edges of \mathcal{E} connecting these vertices.

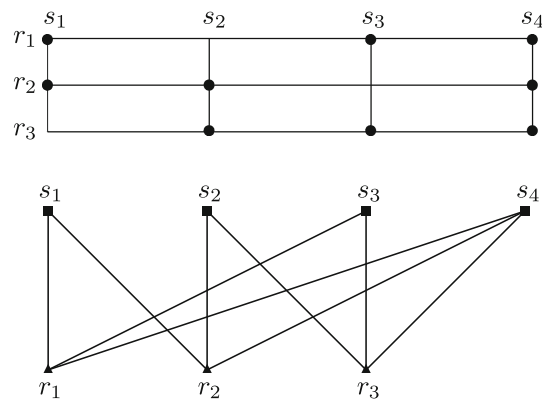


Fig. 1 GNSS grid G and GNSS graph \mathcal{G} . In the scenario described here (Scenario 1), the observational grid G_o includes 12 points: $m = 3$, $n = 4$. The GNSS grid of epoch k , $G \equiv G_k$, includes 9 points; these points are shown as black dots. The corresponding graph, $\mathcal{G} \equiv \mathcal{G}_k$, includes 7 vertices and 9 edges: $n_v = m + n = 7$, $n_e = 9$. The data corresponding to the receiver–satellite pairs (r_1, s_2) , (r_2, s_3) , and (r_3, s_1) are missing

4.2 GNSS spanning tree and loops

As illustrated in Fig. 2, a spanning tree of $\mathcal{G} \equiv \mathcal{G}(\mathcal{V}, \mathcal{E})$ is a subgraph $\mathcal{G}_{st} \equiv \mathcal{G}(\mathcal{V}, \mathcal{E}_{st})$ formed by n_v vertices and $n_v - 1$ 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,

$$c(\ell) := (r_{i_\ell}, s_{j_\ell}), \tag{27}$$

are said to be ‘loop-closure edges’ (see Fig. 2). Their number is denoted by n_c :

$$n_c = n_e - n_{st} \tag{28}$$

where

$$n_{st} := n_v - 1 = m + n - 1 \tag{29}$$

To select a GNSS spanning tree, the edges of \mathcal{E} are first ordered somehow. The corresponding sequence is of the form $\{e(q) : q = 1, \dots, n_e\}$. The algorithm is the following: set $q = 0, n_{st} = 0$, and $\mathcal{E}_{st} = \emptyset$ (the empty set). Then,

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

By construction, the spanning tree thus found depends on how the edges are ordered. The subgrid of G corresponding

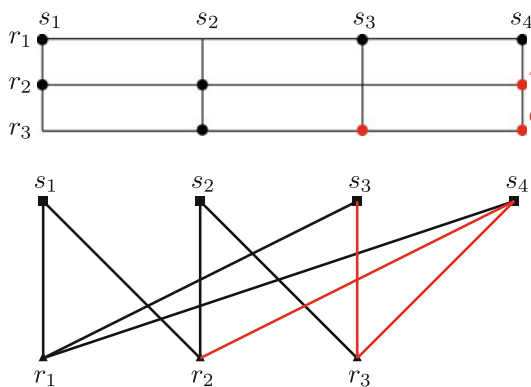


Fig. 2 GNSS spanning tree and loops. The black edges of the graph \mathcal{G} introduced in Fig. 1 are the edges of the selected spanning tree \mathcal{G}_{st} . The points of the corresponding subgrid G_{st} are shown as black dots. The remaining points of G (the red dots of G) correspond to the loop-closure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

to the edges of \mathcal{E}_{st} is denoted by G_{st} ; G_c is that corresponding to the loop-closure edges:

$$G_c := \{(i, j) \in G : (i, j) \notin G_{st}\} \tag{30}$$

Clearly, G_c includes n_c loop-closure points; see Eq. (28) and Fig. 2.

Example 4.1 To show, in concrete manner, how this algorithm works, we now consider its action on the grid G of Fig. 2, 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 three edges of \mathcal{E}_{st} :

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

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 a loop-closure edge. We therefore set

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

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

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

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 [see Eq. (27)]:

$$c(1) = (r_2, s_4)$$

The corresponding loop, (r_2, s_4, r_1, s_1) , is of order 4: it includes four edges (see Fig. 2).

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

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

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:

$$c(2) = (r_3, s_3), \quad c(3) = (r_3, s_4)$$

These loops are of order 6; see Fig. 2.

Remark 4.1 In the special case of the graph shown in Fig. 2, there exist particular spanning trees for which the three loops are of order 4. As the choice of the spanning tree is arbitrary, it is not useful to search for such spanning trees. The connection with the traditional notion of double difference is specified in Sect. 10.

Remark 4.2 In Example 4.1, the points of G are ordered line by line. In fact, to handle some graph transitions (i.e., some scenario changes), one may be led to order them in a more subtle manner; see Sect. 7.4.3 in Lannes and Gratton (2009).

4.3 Reference spaces

For each frequency ν , we now introduce several spaces: the vertex-bias space, the edge-delay space, the spanning-tree space, the closure-delay space, and the bias-delay space.

Vertex-bias space. Let V_0 be the space of functions

$$\alpha := (\alpha^{[r]}, \alpha^{[s]}) \tag{31}$$

where $\alpha^{[s]}(1) = 0$. This space is referred to as the vertex-delay space. Note that [see Eq. (29)]

$$\dim V_0 = n_{st} \tag{32}$$

Edge-delay space. A function ϑ taking its values on G , and thereby on \mathcal{E} , can be regarded as a vector of

$$E := \mathbb{R}^{n_e}$$

The values of ϑ on G are then regarded as the components of ϑ in the standard basis of this edge-delay space.

Spanning-tree and closure-delay spaces. The functions of E that vanish on G_c form a subspace of E denoted by F^{st} : the spanning-tree space. Likewise, the functions of E that vanish on G_{st} form a subspace of E denoted by F^{cd} : the closure-delay (CD) space; this terminology is clarified in Sect. 5. As illustrated in Fig. 3, E is the orthogonal sum of F^{st} and F^{cd} . Clearly [see Eqs. (29) and (28)],

$$\dim F^{st} = n_{st}, \quad \dim F^{cd} = n_c \tag{33}$$

Bias-delay space. By definition, the vertex-bias operator is the operator from V_0 into E defined by the relation (see Remark 4.3 further on)

$$(B\alpha)(i, j) := \alpha^{[r]}(i) - \alpha^{[s]}(j) \tag{34}$$

The range of B , which is denoted by F (see Fig. 3), is referred to as the bias-delay space. Its functions $\beta(i, j)$ are of the form $\alpha^{[r]}(i) - \alpha^{[s]}(j)$. As B is of full rank, we have

$$\dim F = n_{st} \tag{35}$$

Remark 4.3 The matrix of B is generally expressed in the standard bases of V_0 and E . For example in Scenario 1 (see Fig. 1), the points of G being ordered line by line, we then

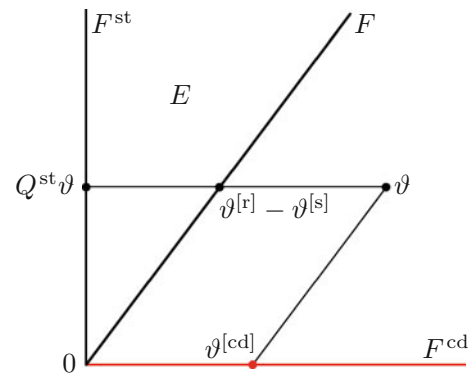


Fig. 3 Canonical decompositions of the edge-delay space. In this geometrical representation of the edge-delay space E , F^{st} is the spanning-tree space. This space is isomorphic to the vertex-bias space V_0 ; $Q^{st}\vartheta$ is the orthogonal projection of ϑ on F^{st} . The orthogonal complement of F^{st} in the Euclidean space E is the closure-delay space F^{cd} . The range of the vertex-bias operator B , the bias-delay space, is a subspace of E denoted by F . This space is isomorphic to F^{st} and thereby to V_0 . As illustrated here, E is the oblique direct sum of F and F^{cd} ; for further details, see Property 2 in Sect. 5.

have

$$[B][\alpha] = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & -1 & 0 \\ 1 & 0 & 0 & 0 & 0 & -1 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & -1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & -1 \\ 0 & 0 & 1 & -1 & 0 & 0 \\ 0 & 0 & 1 & 0 & -1 & 0 \\ 0 & 0 & 1 & 0 & 0 & -1 \end{bmatrix} \begin{bmatrix} \alpha^{[r]}(1) \\ \alpha^{[r]}(2) \\ \alpha^{[r]}(3) \\ \alpha^{[s]}(2) \\ \alpha^{[s]}(3) \\ \alpha^{[s]}(4) \end{bmatrix}$$

The columns of $[B]$ then define the standard basis of F .

5 Approach 1

This approach is based on the following property:

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

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

such that $\vartheta(i, j) = \vartheta^{[r]}(i) - \vartheta^{[s]}(j)$ on the points of \mathcal{G}_{st} . When ϑ is an integer-valued function, these delays are also integer valued.

More concretely, the following process provides these delays in a recursive manner. This type of recursive process was

introduced for the first time in the analysis of the phase calibration operation in PCI; see Sect. 2E in Lannes (2005).

Recursive process. Set $\vartheta^{[s]}(1) = 0$; then span the points of G_{st} line by line (see Fig. 2). 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 $\vartheta^{[s]}(j) = \vartheta^{[r]}(i) - \vartheta(i, j)$.

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

The only operations involved in this process are algebraic sums. The last assertion of Property 1 results from this fact.

Example 5.1 To illustrate this recursive differential process, we now follow its action on the grid G_{st} of Fig. 2. As $\vartheta^{[s]}(1)$ is zero, we then obtain successively:

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

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

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

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

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

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

Closure delays. According to Property 1, the quantities

$$\vartheta^{[cd]}(i, j) := \vartheta(i, j) - [\vartheta^{[r]}(i) - \vartheta^{[s]}(j)] \tag{36}$$

vanish on the points of G_{st} . The values of $\vartheta^{[cd]}(i, j)$ of interest are therefore defined on the CD subgrid G_c ; see Eq. (30) and Fig. 2. We now explain why these n_c values can be referred to as the ‘closure delays’ of ϑ , hence the notation cd or CD.

Let us consider the value of $\vartheta^{[cd]}$ on the ℓ th loop-closure point: $\vartheta^{[cd]}(i_\ell, j_\ell)$; see Eq. (27). Any function β in F satisfies the following property [see Eq. (34)]: the alternating sum of the values of β along the edges of any loop of \mathcal{G} is zero. It then follows from Eq. (36) that $\vartheta^{[cd]}(i_\ell, j_\ell)$ is the alternating sum of the values of ϑ along the edges of the ℓ th loop. More precisely, this result holds when this sum starts with the value of ϑ on the ℓ th loop-closure point. This is why $\vartheta^{[cd]}(i_\ell, j_\ell)$ can be regarded as the closure delay of ϑ for the ℓ th loop. The notion of closure delay therefore generalizes that of double difference. For example, in the scenario of Fig. 2, we have

$$\vartheta^{[cd]}(2, 4) = \vartheta(2, 4) - \vartheta(1, 4) + \vartheta(1, 1) - \vartheta(2, 1)$$

$$\begin{aligned} \vartheta^{[cd]}(3, 3) &= \vartheta(3, 3) - \vartheta(1, 3) + \vartheta(1, 1) - \vartheta(2, 1) \\ &\quad + \vartheta(2, 2) - \vartheta(3, 2) \end{aligned}$$

$$\begin{aligned} \vartheta^{[cd]}(3, 4) &= \vartheta(3, 4) - \vartheta(1, 4) + \vartheta(1, 1) - \vartheta(2, 1) \\ &\quad + \vartheta(2, 2) - \vartheta(3, 2) \end{aligned}$$

The following property, which is a simple transcription of Eq. (36), therefore solves the decomposition problem (21):

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

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

For a given spanning tree, this decomposition is unique. When ϑ is an integer-valued function, $\vartheta^{[r]}$, $\vartheta^{[s]}$, and $\vartheta^{[cd]}$ are also integer valued. In particular,

$$N_v(i, j) = [N_v^{[r]}(i) - N_v^{[s]}(j)] + N_v^{[cd]}(i, j)$$

Note that the uniqueness of this decomposition results from Property 1.

As illustrated in Fig. 3, $\vartheta^{[cd]}$ is the oblique projection of ϑ on F^{cd} along F . The corresponding operator is the ‘closure operator’ C :

$$\vartheta^{[cd]} = C\vartheta \tag{37}$$

Its null space (i.e., its kernel) is the range of B :

$$\ker C = F \tag{38}$$

According to Property 2, it is also clear that E is the direct sum of F and F^{cd} (see Fig. 3):

$$E = F \oplus F^{cd} \quad (F \cap F^{cd} = \{0\}) \tag{39}$$

Let us now denote by $[C]$ the matrix of the closure operator expressed in the standard basis of E . The entries of the column vector of $[C]$ corresponding to a loop-closure point are zero, except that of the corresponding CD component. Clearly, that entry is equal to unity. The entries of the column vector of $[C]$ corresponding to a spanning-tree point are obtained via Eq. (36) which then yields

$$\vartheta^{[cd]}(i_\ell, j_\ell) = -[\vartheta^{[r]}(i_\ell) - \vartheta^{[s]}(j_\ell)]$$

for $\ell = 1, \dots, n_c$; the other entries are zero. The delays $\vartheta^{[r]}(i_\ell)$ and $\vartheta^{[s]}(j_\ell)$ are then provided by the recursive process in which ϑ is the characteristic function of that spanning-tree point: ϑ vanishes on G except on that point for which $\vartheta = 1$; see Example 5.1. Clearly, the edges of the closure loops are identified via the nonzero entries of the lines of $[C]$.

6 Approach 2

In that approach, one concentrates on the rank defect of the reference problem defined in the context of Eq. (21). This rank defect is equal to n_{st} . Indeed,

$$(n_{st} + n_e) - n_e = n_{st}$$

The simplest way to correct for this defect is to impose the minimum constraint ‘ $\gamma = 0$ on G_{st} ’ (see Teunissen 1984). One is then led to solve the problem of full rank

$$\alpha^{[r]}(i) - \alpha^{[s]}(j) = \vartheta(i, j), \quad \text{for all } (i, j) \in G_{st} \tag{40}$$

i.e.,

$$Q^{st} B \alpha = Q^{st} \vartheta \tag{41}$$

where Q^{st} is the operator from E onto F^{st} induced by the orthogonal projection (operator) of E onto F^{st} ; see Fig. 3. Expressed in the standard bases of V_0 and F^{st} , Eq. (41) can be written in the matrix form

$$[Q^{st}][B][\alpha] = [Q^{st}][\vartheta] \tag{42}$$

where

$$[\alpha] := [\alpha^{[r]}(1) \dots \alpha^{[r]}(m) \alpha^{[s]}(2) \dots \alpha^{[s]}(n)]^T \tag{43}$$

Denoting by α_ϑ the solution associated with ϑ , we therefore have

$$[\alpha_\vartheta] = ([Q^{st}][B])^{-1} [Q^{st}][\vartheta] \tag{44}$$

Clearly, $[Q^{st}][B]$ is obtained by removing from $[B]$ the lines corresponding to the loop-closure points of G . For example, in the case of Fig. 2, the points of G_{st} being ordered line by line, we have (see Remark 4.3)

$$[Q^{st}][B] = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & -1 & 0 \\ 1 & 0 & 0 & 0 & 0 & -1 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & -1 & 0 & 0 \\ 0 & 0 & 1 & -1 & 0 & 0 \end{bmatrix}$$

For clarity, the receiver and satellite components of α_ϑ are denoted by $\vartheta^{[r]}$ and $\vartheta^{[s]}$, respectively:

$$[\alpha_\vartheta] \equiv [\vartheta^{[r]}(1) \dots \vartheta^{[r]}(m) \vartheta^{[s]}(2) \dots \vartheta^{[s]}(n)]^T \tag{45}$$

The oblique projection of E onto F along F^{cd} is then given by the relation (see Fig. 3)

$$[B][\alpha_\vartheta] = [B] ([Q^{st}][B])^{-1} [Q^{st}][\vartheta] \tag{46}$$

As

$$(B\alpha_\vartheta)(i, j) = \vartheta^{[r]}(i) - \vartheta^{[s]}(j) \tag{47}$$

The minimum constrained solution in γ is therefore given by the formula [see Eq. (21)]

$$\gamma_\vartheta(i, j) = \vartheta(i, j) - [\vartheta^{[r]}(i) - \vartheta^{[s]}(j)] \tag{48}$$

Setting

$$\vartheta^{[cd]}(i, j) := \gamma_\vartheta(i, j) \tag{49}$$

we then retrieve Eq. (36), and thereby Property 2 in particular. The equivalence of Approaches 1 and 2 is thus established.

In Approach 2, the matrix of C is explicitly given by the formula [see Eq. (46)]

$$[C] = [I] - [B] ([Q^{st}][B])^{-1} [Q^{st}] \tag{50}$$

Here, $[I]$ is the identity on E . The entries of the lines of $[C]$ corresponding to the points of G_{st} are a priori zero. In other words, they must not be computed.

Remark 6.1 It can be shown that $[Q^{st}][B]$ is a unimodular matrix; see Example 1 in Appendix 1 and A1.2–A1.4 (By definition, a unimodular matrix is a square integer matrix with determinant ± 1). The inverse of $[Q^{st}][B]$ can then be computed via the integer-programming technique described in Appendix A1.4; see Example 4 in Appendix 1. This pointed out, the recursive process of Approach 1 provides the column vectors of $([Q^{st}][B])^{-1}$ in a simpler manner; see how $[C]$ is built in Sect. 5.

Remark 6.2 In the S -basis technique of Baarda (1973), Teunissen (1984), and de Jonge (1998), S^\perp is then the matrix whose column vectors form the standard basis of F^{st} ; $[Q^{st}]$ is therefore the transpose of matrix S^\perp : $[Q^{st}] \equiv S^{\perp T}$. Likewise, $[B]$ corresponds to some choice of ‘ V ’ in the notation of de Jonge. Here, the columns of $[B] \equiv V$ form the standard basis of F . This explicitly shows that $[C]$ is the S -transformation of the S -basis technique; see Sect. 3.4 in de Jonge (1998). The ‘estimable functions of carrier-wave ambiguities’ are therefore the CD ambiguities $N_v^{[cd]}(i_\ell, j_\ell)$.

Remark 6.3 To handle some graph transitions, one may be led to change the selected spanning tree; see Remark 4.2. Denoting by C_1 and C_2 the oblique projections associated with these spanning trees, we then have $C_2 = C_2 C_1$. Indeed, C_1 and C_2 have the same null space: F . As a result, for any minimum constrained solution $\vartheta_1 := C_1 \vartheta$, we have $\vartheta_2 = C_2 \vartheta_1$. One thus passes from a minimum constrained solution to another by some S -transformation. Explicitly [see Eq. (50)],

$$\vartheta_2 = \vartheta_1 - [B] ([Q_2^{st}][B])^{-1} [Q_2^{st}][\vartheta_1]$$

Remark 6.4 The matrix of C can also be represented in the form

$$[C] = [K]([L]^T[K])^{-1}[L]^T$$

in which $[K]$ and $[L]$ are matrices whose column vectors form bases of F^{cd} and F^\perp , respectively. To get $[C]$, the matrix to be inverted is then of dimension $n_c \times n_c$, instead of $n_{st} \times n_{st}$ in Eq. (50).

7 Canonical equations

According to Property 2, the undifferenced equation (10) can be finally parametrized in the canonical form

$$\Phi_{v,k}^*(i, j) = [\tilde{\alpha}_{\phi;v,k}^{[r]}(i) - \tilde{\alpha}_{\phi;v,k}^{[s]}(j)] + \lambda_v N_v^{[cd]}(i, j) + \varepsilon_{\phi;v,k}(i, j) \tag{51}$$

where

$$\tilde{\alpha}_{\phi;v,k}^{[r]}(i) := \alpha_{\phi;v,k}^{[r]}(i) + \lambda_v N_v^{[r]}(i) \tag{52}$$

$$\tilde{\alpha}_{\phi;v,k}^{[s]}(j) := \alpha_{\phi;v,k}^{[s]}(j) + \lambda_v N_v^{[s]}(j) \tag{53}$$

Concerning the phase, $\tilde{\alpha}_{\phi;v,k}^{[r]}$, $\tilde{\alpha}_{\phi;v,k}^{[s]}$, and $N_v^{[cd]}$ are the ‘estimable functional variables’ of the problem. In particular, according to Eqs. (6), (12), (14) and (53), the estimable phase satellite-clock error is then the frequency-dependent phase satellite-clock error biased by:

- a tropospheric satellite delay;
- an ionospheric satellite delay;
- an initial satellite phase;
- an integer satellite ambiguity.

The canonical form of the undifferenced code equation is nothing but Eq. (15):

$$P_{v,k}^*(i, j) = [\alpha_{p;v,k}^{[r]}(i) - \alpha_{p;v,k}^{[s]}(j)] + \varepsilon_{p;v,k}(i, j) \tag{54}$$

From Eqs. (9), (17), and (19), the estimable code satellite-clock error is then the original frequency-dependent code satellite-clock error biased by:

- a tropospheric satellite delay;
- an ionospheric satellite delay.

Remark 7.1 When one is not interested in the receiver functional variables $\tilde{\alpha}_{\phi;v,k}^{[r]}$ and $\alpha_{p;v,k}^{[r]}$, one may be led to consider the single-difference (SD) equations induced by the ‘canonical undifferenced equations’ (51) and (54). For example, for each frequency, by spanning the GNSS grid G line by line (see Fig. 2), the phase equation (51) then yields $(n_e - n_r)$ SD equations in which the receiver biases $\tilde{\alpha}_{\phi;v,k}^{[r]}(i)$ no longer appear. The CD ambiguities of course remain involved in these ‘canonical SD equations.’ For quality control, it is however preferable to work with the canonical undifferenced equations. Indeed, in the DIA methods (see, e.g., Lannes and Gratton 2009), most of the outliers can then be identified more directly.

8 Solution of the problem: survey

For our present purposes, let us consider an observational period $[1, k]$ where no receiver–satellite signal appears or disappears. Over that period, the problem to be solved in the LS sense is then governed by a system of linear equations of the form

$$\begin{cases} A_1 u_1 + Bv = b_1 \\ A_2 u_2 + Bv = b_2 \\ \vdots \\ A_k u_k + Bv = b_k \end{cases} \tag{55}$$

where A_k and B are linear operators. With regard to the canonical equations (51) and (54), the functional variables $\tilde{\alpha}_{\phi;v,k}^{[r]}$, $\tilde{\alpha}_{\phi;v,k}^{[s]}$, $\alpha_{p;v,k}^{[r]}$, and $\alpha_{p;v,k}^{[s]}$ are then the components of the local variable u_k . When one works in canonical SD mode, the receiver pseudo-clock variables no longer appear; see Remark 7.1. In both cases, the entries of the global variable v are the integer CD ambiguities $N_v^{[cd]}(i_\ell, j_\ell)$.

The number of observations must be greater than or equal to the number of unknown parameters; this is a necessary condition for solving the problem. At epoch k , with n_v carrier waves, we must therefore have, with regard to Eqs. (51) and (54),

$$2kn_v n_e \geq 2kn_v n_{st} + n_v n_c$$

i.e., since $n_c = n_e - n_{st}$ [see Eq. (28)],

$$2k \geq 1$$

This condition is therefore always satisfied. In fact, the preliminary parametrization performed in Sect. 2 was completed in Sect. 7 so that the problem is then of full rank. Equation (55) can then be solved in the LS sense, and recursively, by using for example the QR method (see Lannes and Gratton 2009). Other techniques can of course be implemented (see, e.g., de Jonge 1998). At each epoch k , one thus obtains, in particular, the float ambiguity \hat{v}_k and the Cholesky factor R_k of the inverse of its variance–covariance matrix. This upper-triangular matrix is then decorrelated via appropriate techniques based on the LLL algorithm (see, Lannes and Gratton 2009), or on the LAMBDA method (see Teunissen 1995). The integer ambiguity solution \check{v}_k is then obtained by using classical integer-programming techniques. Once \check{v}_k has been fixed to some \check{v} (“ $\check{v}_k \rightarrow \check{v}$ ”), and \check{v} has been validated (see Verhagen and Teunissen 2006), the problem can be completely solved. We then have $\check{v}_v = N_v^{[cd]}$ for all v . The corresponding estimate of the real-valued functional variable u_k is then obtained. For clarity, the latter is denoted by \check{u}_k . In the following section, the estimates of $\tilde{\alpha}_{\phi;v,k}^{[s]}(j)$ and $\alpha_{p;v,k}^{[s]}(j)$ thus obtained, $\check{\alpha}_{\phi;v,k}^{[s]}(j)$ and $\check{\alpha}_{p;v,k}^{[s]}(j)$, are simply referred to as the satellite pseudo-clock biases.

9 Related PPP equations

The satellite pseudo-clock biases $\check{\alpha}_{\phi;v,k}^{[s]}(j)$ and $\check{\alpha}_{p;v,k}^{[s]}(j)$ for $j \neq 1$ can be broadcasted to the network users for their precise point positioning (PPP). As clarified in this section, the form of the equations to be solved by the user is then similar to that of the traditional PPP equations. The corresponding standard approaches are described in [Zumberge et al. \(1997\)](#), [Ge et al. \(2008\)](#), and [Bertiger et al. \(2010\)](#).

Let us denote by r_t the user receiver. According to Eqs. (5) and (10), the user phase equation is the following:

$$\Phi_{v,k}(t, j) = \rho_k(t, j) + [\alpha_{\phi;v,k}^{[r]}(t) - \alpha_{\phi;v,k}^{[s]}(j)] + \lambda_v N_v(t, j) + \varepsilon_{\phi;v,k}(t, j) \tag{56}$$

Concerning the code, we then have

$$P_{v,k}(t, j) = \rho_k(t, j) + [\alpha_{p;v,k}^{[r]}(t) - \alpha_{p;v,k}^{[s]}(j)] + \varepsilon_{p;v,k}(t, j) \tag{57}$$

The receiver–satellite range can be linearized in the form

$$\rho_k(t, j) = \rho_k^0(t, j) + \sum_{p=1}^3 c_k^{(p)}(t, j) \xi_{t,k}^{(p)} \tag{58}$$

where $\rho_k^0(t, j)$ is the nominal value of $\rho_k(t, j)$; $\xi_{t,k}^{(p)}$ is the p th (position) increment of the user receiver at epoch k . Note that $c_k^{(p)}(t, j)$ is a direction-cosine function [see, e.g., Eq. (14) in [Lannes and Gratton \(2008\)](#)].

Taking into account Eq. (53), we have (with regard to the selected spanning tree of the GNSS network graph)

$$\begin{cases} \alpha_{\phi;v,k}^{[s]}(j) \simeq \check{\alpha}_{\phi;v,k}^{[s]}(j) - \lambda_v N_v^{[s]}(j) \\ \alpha_{p;v,k}^{[s]}(j) \simeq \check{\alpha}_{p;v,k}^{[s]}(j) \end{cases} \tag{59}$$

We are then led to set

$$\begin{cases} \Phi_{v,k}^{(i)}(j) := \Phi_{v,k}(t, j) - \rho_k^0(t, j) + \check{\alpha}_{\phi;v,k}^{[s]}(j) \\ P_{v,k}^{(i)}(j) := P_{v,k}(t, j) - \rho_k^0(t, j) + \check{\alpha}_{p;v,k}^{[s]}(j) \end{cases} \tag{60}$$

and

$$N_v^{(i)}(j) := N_v(t, j) + N_v^{[s]}(j) \tag{61}$$

Equations (56) and (57) then yield the PPP equations

$$\begin{aligned} \Phi_{v,k}^{(i)}(j) &= \sum_{p=1}^3 c_k^{(p)}(t, j) \xi_{t,k}^{(p)} + \alpha_{\phi;v,k}^{[r]}(t) \\ &+ \lambda_v N_v^{(i)}(j) + \bar{\varepsilon}_{\phi;v,k}(t, j) \end{aligned} \tag{62}$$

$$\begin{aligned} P_{v,k}^{(i)}(j) &= \sum_{p=1}^3 c_k^{(p)}(t, j) \xi_{t,k}^{(p)} + \alpha_{p;v,k}^{[r]}(t) \\ &+ \bar{\varepsilon}_{p;v,k}(t, j) \end{aligned} \tag{63}$$

As $\check{\alpha}_{\phi;v,k}^{[s]}$ and $\check{\alpha}_{p;v,k}^{[s]}$ are estimates of $\tilde{\alpha}_{\phi;v,k}^{[s]}$ and $\alpha_{p;v,k}^{[s]}$ respectively, $\bar{\varepsilon}_{\phi;v,k}$ and $\bar{\varepsilon}_{p;v,k}$ differ from $\varepsilon_{\phi;v,k}$ and $\varepsilon_{p;v,k}$, respectively. To solve the problem, the variance covariance matrices of $\bar{\varepsilon}_{\phi;v,k}$ and $\bar{\varepsilon}_{p;v,k}$ are then to be properly taken into account. Estimates of the increments $\xi_{t,k}^{(p)}$ can thus be obtained.

Remark 9.1 The form of the PPP equations (62) and (63) is similar to that of the standard PPP equations. Here, however, in the case of relatively small networks, the tropospheric and ionospheric delays no longer appear explicitly. In fact, they are taken into account via the definitions of $\Phi_{v,k}^{(i)}(j)$ and $P_{v,k}^{(i)}(j)$; see Eqs. (6), (9), (12), (14), (17), (19), (53), and (60). Clearly, for the PPP user, the network plays the role of a calibrator; see Remark 3.1. This pointed out, we chose for small networks only because of the ease with which our approach can then be presented and explained.

10 Connection with the DD approach

We first introduce the notion of DD operator (Sect. 10.1). We then analyze the relationship between the DDs and the CDs (Sect. 10.2). In particular, we then show that the information provided by a maximum set of independent DDs may not reach that of a complete set of CDs. The methods based on the CD approach are therefore preferable to those deriving from the DD approach. The related points are specified in Sects. 10.3 and 10.4.

10.1 Notion of DD operator

The usual way of getting rid of the receiver and satellite biases is to form double differences. Given some ϑ in E , the latter are of the form

$$\begin{aligned} &[\vartheta(i_1, j_1) - \vartheta(i_1, j_2)] - [\vartheta(i_2, j_1) - \vartheta(i_2, j_2)] \\ &= \vartheta(i_1, j_1) - \vartheta(i_1, j_2) + \vartheta(i_2, j_2) - \vartheta(i_2, j_1) \end{aligned}$$

Indeed, we then have

$$\begin{aligned} &[\alpha^{[r]}(i_1) - \alpha^{[s]}(j_1)] - [\alpha^{[r]}(i_1) - \alpha^{[s]}(j_2)] \\ &+ [\alpha^{[r]}(i_2) - \alpha^{[s]}(j_2)] - [\alpha^{[r]}(i_2) - \alpha^{[s]}(j_1)] = 0 \end{aligned}$$

Clearly, the corresponding loop of order 4 is characterized by the points (i_1, j_1) and (i_2, j_2) of G . The maximum number of such loops, n_{dd} , depends on the scenario under consideration. For example, in Scenario 1 (see Fig. 1), we have $n_{dd} = 3$. These loops are then characterized by the

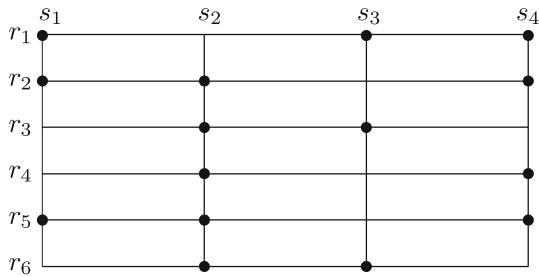


Fig. 4 Scenario 2. In the scenario considered here, G includes 15 points: $n_e = 15$. The number of loops of order 4, and thereby, the number of DDs is equal to 8: $n_{dd} = 8$. The maximum number of independent DDs proves then to be equal to 5: $n_d^m = 5$; see text. The number of CDs is then equal to 6: $n_c = 15 - 9$; for further details concerning the CD approach, see Fig. 5. Here, n_d^m is therefore strictly less than n_c : the information provided by a maximum set of independent DDs does not reach that of a complete set of CDs. In this scenario, this property results from the fact that no loop of order 4 includes edge (r_1, s_3) . In the traditional DD approach, the data associated with this edge are not used

following pairs of points:

- [(1, 1), (2, 4)]
- [(1, 3), (3, 4)]
- [(2, 2), (3, 4)]

Let us order the points of G line by line. The matrix of the operator from E into \mathbb{R}^3 that provides all the DDs is then the following:

$$[D] = \begin{bmatrix} \mathbf{1} & 0 & -1 & -1 & 0 & 1 & 0 & 0 & 0 \\ 0 & \mathbf{1} & -1 & 0 & 0 & 0 & 0 & -1 & 1 \\ 0 & 0 & \mathbf{0} & \mathbf{0} & \mathbf{1} & -1 & -1 & 0 & 1 \end{bmatrix}$$

By definition, such a matrix has n_e columns (here, 9). In this case, the lines of $[D]$ are linearly independent. One then says that the DDs thus defined are linearly independent. Here, the maximum number of independent DDs is therefore equal to 3:

$$n_d^m = 3$$

Scenario 1 therefore corresponds to a situation where $n_d^m = n_c$. This is not always the case.

To illustrate this point in an elementary manner, let us consider the scenario of Fig. 4: Scenario 2; n_{dd} is then equal to 8. The following pairs of points characterize these loops:

- [(1, 1), (2, 4)]
- [(1, 1), (5, 4)]
- [(2, 1), (5, 2)]
- [(2, 1), (5, 4)]
- [(2, 2), (4, 4)]
- [(2, 2), (5, 4)]
- [(3, 2), (6, 3)]
- [(4, 2), (5, 4)]

The matrix $[D]$ of the operator that provides all the DDs has then 8 lines and 15 columns:

$$\begin{bmatrix} 1 & 0 & -1 & -1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & -1 & 0 & 0 & 0 & 0 & 0 & -1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & -1 & 0 & 0 & 0 & 0 & -1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & -1 & 0 & 0 & -1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & -1 & 0 & 0 & 0 & 0 & 0 & -1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & -1 & 0 & 0 & 0 & 0 & 0 & -1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & -1 & 0 & -1 & 1 & 0 & 0 \end{bmatrix}$$

It is easy to check that the DDs defined by this matrix are not linearly independent: they do not form a ‘free set.’ In fact, as clarified below, n_d^m is then equal to 5.

To find a maximum set of independent DDs, several methods can be implemented. Some of these appeal to standard algebraic techniques (see, e.g., Björck 1996). Others are based on Boolean methods (see Saalfeld 1999). Applied to the complete matrix $[D]$, the founding algorithm described in Appendix A1.3 also provides such a set; see Example 3 in Appendix A1.3.

In the special case under consideration (Scenario 2), the following loops of order 4 form a maximum set of independent DDs:

- [(1, 1), (2, 4)]
- [(1, 1), (5, 4)]
- [(2, 1), (5, 2)]
- [(3, 2), (6, 3)]
- [(2, 2), (4, 4)]

Any matrix $[D]$ that provides a set of independent DDs is said to be a DD matrix. The number of lines of $[D]$ is then denoted by n_d . We therefore have

$$n_d \leq n_d^m \leq n_{dd} \tag{64}$$

In the DD approaches, it is therefore recommended to work with a maximum set of independent DDs. For example, with regard to Scenario 2, the following operator provides such a set (see Example 3 in Appendix A1.3):

$$[D] = \begin{bmatrix} 1 & 0 & -1 & -1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & -1 & 0 & 0 & 0 & 0 & 0 & -1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & -1 & 0 & 0 & 0 & 0 & 0 & -1 & 1 \\ 0 & 0 & 0 & 0 & 1 & -1 & 0 & 0 & -1 & 1 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

According to basic notions of linear algebra, n_d columns of $[D]$ at most can be linearly independent: the rank of D (the dimension of its range) is equal to n_d . The dimension of the null space of D is therefore equal to $n_e - n_d$. We thus have [see Eqs. (28), (29), and (33)]

$$\begin{aligned} \dim(\ker D) &= (n_{st} + n_c) - n_d \\ &= \dim F + (n_c - n_d) \end{aligned} \tag{65}$$

Furthermore, D annihilates the vertex biases: $D\beta = 0$ for any β in F ; we therefore have

$$F \subseteq \ker D \quad (F = BV_0 = \ker C) \tag{66}$$

Consequently, $\dim F \leq \dim(\ker D)$. It then follows from Eq. (65) that

$$n_c - n_d \geq 0 \tag{67}$$

The following inequality therefore holds:

$$n_c - n_d^m \geq 0 \tag{68}$$

For example, in Scenarios 1 and 2, we respectively have $n_d^m = n_c$ and $n_d^m < n_c$. This explicitly shows that the information provided by a maximum set of independent DDs may not reach that of a complete set of CDs; see, for example, Figs. 4, 5, and 7. In that case, as F is a subspace of the null space of D , double differencing annihilates more than the pseudo-clock biases.

10.2 The DD–CD relationship

Let D^{cd} be the operator from F^{cd} into \mathbb{R}^{n_d} induced by D . Its matrix, $[D^{cd}]$ includes n_d lines and n_c columns. This matrix is obtained by removing from $[D]$ the columns corresponding to the spanning-tree points of G . The columns of $[D^{cd}]$ are therefore associated with the loop-closure points ordered line by line. For example, in Scenario 2 (see Sect. 10.1), we have

$$[D^{cd}] = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & -1 & 0 & 0 & 0 & 1 \\ -1 & 0 & 1 & 0 & 0 & 0 \end{bmatrix}$$

Let us now denote by K the null space of D^{cd} . From the analysis developed in Sect. 10.1,

$$\dim K = n_c - n_d \tag{69}$$

The rank of D^{cd} is equal to n_d . The range of D^{cd} is therefore defined by a set of n_d independent column vectors of $[D^{cd}]$. As specified in Appendix A1.3, the best way of finding such a set is to apply elementary unimodular matrices on the left-hand side of $[D^{cd}]$. For example, in Scenario 2, we then get the following matrix:

$$\begin{bmatrix} \mathbf{1} & 0 & 0 & 0 & 0 & 0 \\ 0 & \mathbf{1} & 0 & 0 & 0 & -1 \\ 0 & 0 & \mathbf{1} & 0 & 0 & 0 \\ 0 & 0 & 0 & \mathbf{1} & 0 & 0 \\ 0 & 0 & 0 & 0 & \mathbf{1} & \mathbf{0} \end{bmatrix}$$

The independent column vectors thus identified define a set of $n_d = 5$ loop-closure points. Here, this set includes the following points (see Fig. 5): $c_1 = (2, 4)$, $c_2 = (3, 3)$, $c_3 = (4, 4)$,

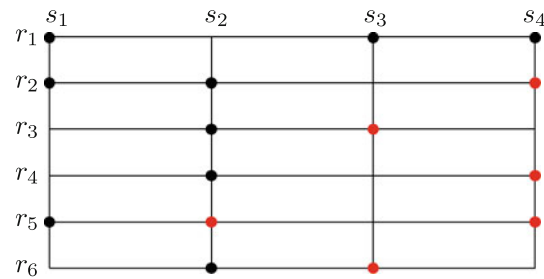


Fig. 5 Selected spanning tree for Scenario 2. This figure corresponds to the CD approach of Scenario 2; see Fig. 4. Here, the selected spanning tree is built by spanning the points of G line by line. The points of G_{st} are shown as black dots. The red ones are the corresponding loop-closure points; see Fig. 2. With regard to the selected spanning tree, we then have three loops of order 4 and three loops of order 6.

$c_4 = (5, 2)$, and $c_5 = (5, 4)$. Let us denote by \mathbb{H} the subset of integers characterizing the order of these points in the list of the loop-closure points. Here,

$$\mathbb{H} = \{1, 2, 3, 4, 5\}$$

The remaining $n_c - n_d$ loop-closure points define a set of integers denoted by \mathbb{L} . Here, this set includes only one element:

$$\mathbb{L} = \{6\}$$

Let us denote by e_ℓ the characteristic function of point c_ℓ :

$$e_\ell(i, j) := \begin{cases} 1, & \text{if } (i, j) = c_\ell; \\ 0, & \text{otherwise} \end{cases} \tag{70}$$

By definition, H is the subspace of F^{cd} generated by the $e_\ell s$, ℓ lying in \mathbb{H} . Likewise, L denotes the subspace generated by the $e_\ell s$, ℓ lying in \mathbb{L} . By construction, L is the orthogonal complement of H in F^{cd} ; see Fig. 6.

The dimension of $D^{cd}H$, n_d , is equal to that of H . The operator of H into \mathbb{R}^{n_d} induced by D^{cd} , D_H^{cd} , is therefore invertible. More precisely (see Appendix A1.4), the matrix of this operator is unimodular. For example in Scenario 2, we have

$$[D_H^{cd}] = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & -1 & 0 & 0 & 0 \\ -1 & 0 & 1 & 0 & 0 \end{bmatrix}$$

and (see Example 5 in Appendix A1.4)

$$[D_H^{cd}]^{-1} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 & 0 \\ 1 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \end{bmatrix}$$

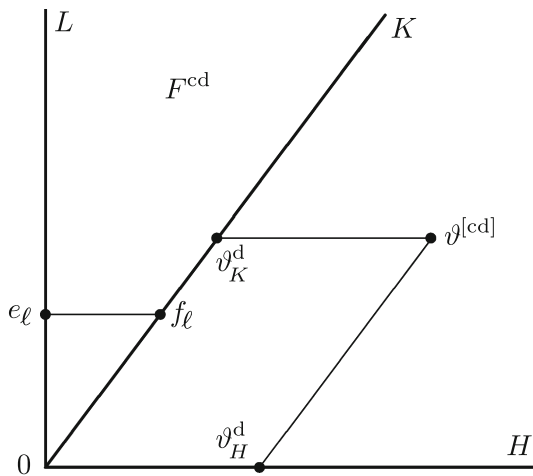


Fig. 6 Decomposition of F^{cd} induced by the DD operator D . In this geometrical representation of the CD space F^{cd} , K is the null space of D^{cd} , the operator from F^{cd} into \mathbb{R}^{n_d} induced by D . As specified in the text, H is a ‘trivial subspace’ of F^{cd} for which the following property is satisfied: the operator from H into \mathbb{R}^{n_d} induced by D^{cd} is invertible. Moreover, its matrix $[D_H^{cd}]$ is unimodular. Via ϑ_H^d , the DD vector defines the corresponding CD vector up to a vector in K ; see Eqs. (71)–(73). Here, L is the orthogonal complement of H in F^{cd} ; K is isomorphic to L . For further details, see Properties 3, 4 and Eq. (77).

Let us now denote by $\theta^{[dd]}$ the DD vector of some ϑ in E :

$$\theta^{[dd]} := D\vartheta = D\vartheta^{[cd]} \quad (\theta^{[dd]} \in \mathbb{R}^{n_d}) \tag{71}$$

Setting

$$\vartheta_H^d := (D_H^{cd})^{-1}\theta^{[dd]} \tag{72}$$

we have

$$D\vartheta_H^d = \theta^{[dd]} \tag{73}$$

As illustrated in Fig. 6, ϑ_H^d is therefore the oblique projection of $\vartheta^{[cd]}$ on H along K . As a result, the oblique projection of $\vartheta^{[cd]}$ on K along H is the vector

$$\vartheta_K^d := \vartheta^{[cd]} - (D_H^{cd})^{-1}\theta^{[dd]} \tag{74}$$

We thus have the following property (see Fig. 6):

Property 3 Any CD function can be decomposed in the form

$$\vartheta^{[cd]} = \vartheta_H^d + \vartheta_K^d$$

where

$$\vartheta_H^d := (D_H^{cd})^{-1}D\vartheta^{[cd]}$$

For a given H , this decomposition is unique. As a corollary, F^{cd} is the direct sum of H and K .

Note that this property, which completes Property 2, is of the same type; compare Fig. 6 with Fig. 3.

The DD vector defines the corresponding CD vector up to a vector of K . In the special case where n_d is equal to n_c , K is

reduced to $\{0\}$; then $H = F^{cd}$ and $\vartheta^{[cd]} = \vartheta_H^d$. The DD vector then defines the CD vector without any ambiguity [see Eq. (72)]:

$$\vartheta^{[cd]} = (D^{cd})^{-1}\theta^{[dd]} \quad (n_d = n_c) \tag{75}$$

For example, in Scenario 1,

$$[D^{cd}] = \begin{bmatrix} 1 & 0 & 0 \\ 0 & -1 & 1 \\ -1 & 0 & 1 \end{bmatrix}, \quad [D^{cd}]^{-1} = \begin{bmatrix} 1 & 0 & 0 \\ 1 & -1 & 1 \\ 1 & 0 & 1 \end{bmatrix}$$

In the general where n_d is strictly less than n_c , K can easily be characterized. We now specify this point.

Let us denote by $\mathbb{Z}(F^{cd})$ the CD lattice. Clearly, $\{e_\ell\}_{\ell=1}^{n_c}$ is the standard basis of this integer lattice. Likewise, define $\mathbb{Z}(H)$ and $\mathbb{Z}(L)$ as the \mathbb{Z} -lattices with bases $\{e_\ell\}_{\ell \in \mathbb{H}}$ and $\{e_\ell\}_{\ell \in \mathbb{L}}$, respectively. By construction, $\mathbb{Z}(F^{cd})$ is the direct sum of $\mathbb{Z}(H)$ and $\mathbb{Z}(L)$:

$$\mathbb{Z}(F^{cd}) = \mathbb{Z}(H) \oplus \mathbb{Z}(L) \tag{76}$$

For any $\ell \in \mathbb{L}$, we now define f_ℓ as the oblique projection of e_ℓ on K along H (see Fig. 6). According to Property 3, we have

$$f_\ell := e_\ell - (D_H^{cd})^{-1}De_\ell \quad (\ell \in \mathbb{L}) \tag{77}$$

As e_ℓ is the projection of f_ℓ on L (see Fig. 6), and $\{e_\ell\}_{\ell \in \mathbb{L}}$ is a basis for L , $\{f_\ell\}_{\ell \in \mathbb{L}}$ is a basis for K . Clearly, K is isomorphic to L .

In the following property, which can be regarded as a corollary of Property 3, $\mathbb{Z}(K)$ is the \mathbb{Z} -lattice with basis $\{f_\ell\}_{\ell \in \mathbb{L}}$. For example, in Scenario 2, $\mathbb{Z}(K)$ is the one-dimensional \mathbb{Z} -lattice generated by the CD vector whose components are the six entries of the following column matrix:

$$[f_6] := \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ \mathbf{1} \end{bmatrix} - \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 & 0 \\ 1 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \\ 0 \end{bmatrix} = \begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \\ 0 \\ 1 \end{bmatrix}$$

Property 4 The CD lattice $\mathbb{Z}(F^{cd})$ is the direct sum of the integer lattices $\mathbb{Z}(H)$ and $\mathbb{Z}(K)$:

$$\mathbb{Z}(F^{cd}) = \mathbb{Z}(H) \oplus \mathbb{Z}(K)$$

In the general case where n_d is strictly less than n_c , we therefore have, from Property 3 and Notation (73),

$$N_v^{[cd]} = (D_H^{cd})^{-1}N_v^{[dd]} + \sum_{\ell \in \mathbb{L}} N_{v;\ell}^{[K]} f_\ell \tag{78}$$

for some ambiguities $N_{v;\ell}^{[K]}$ in \mathbb{Z} . These $n_c - n_d$ integer ambiguities are referred to as the K -ambiguities. Clearly, $N_v^{[dd]}$ includes n_d components: the integer DD ambiguities $N_{v;q}^{[dd]}$.

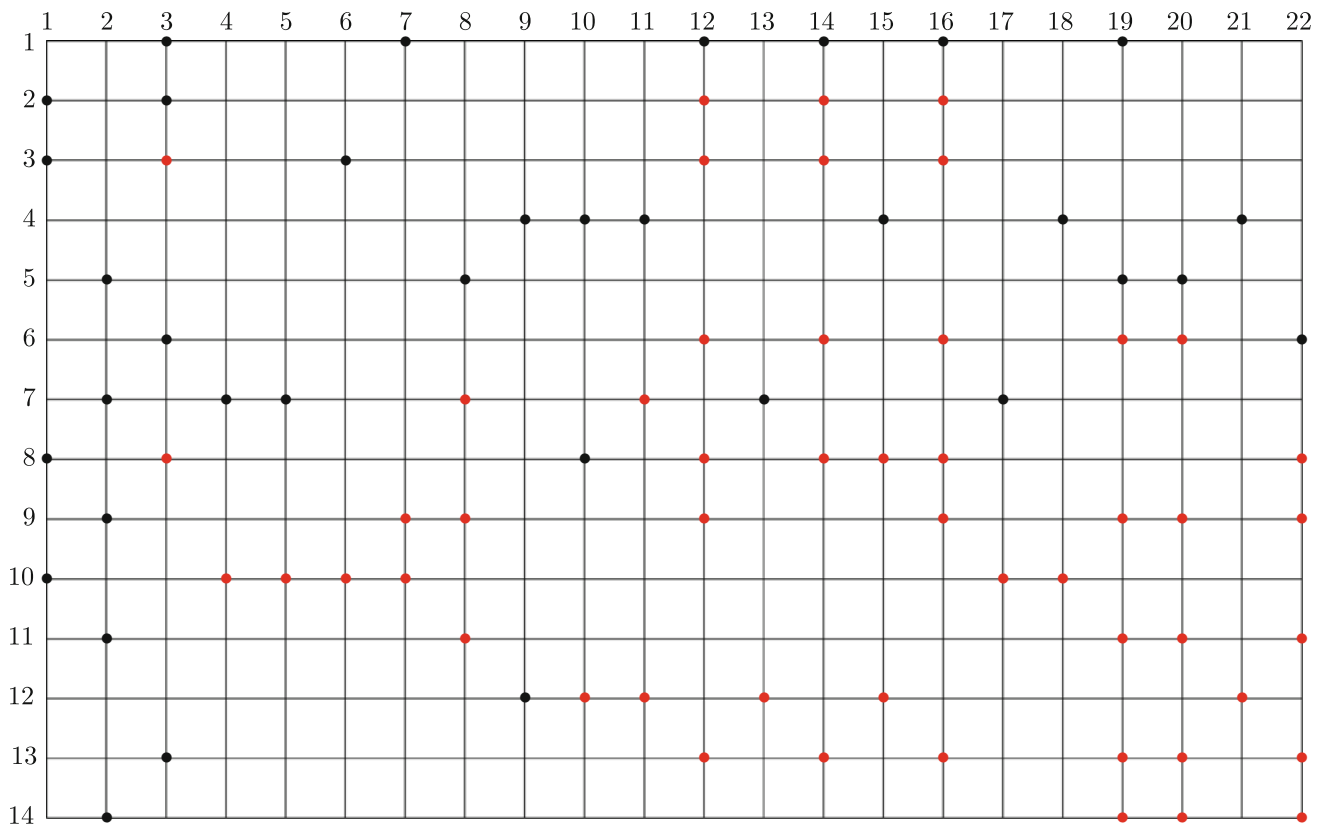


Fig. 7 Scenario 3. This example corresponds to the scenario defined in Table 3.2 of de Jonge (1998); $m = 14$, $n = 22$, and $n_c = 86$. The spanning tree of the corresponding graph is built by spanning the points of G line by line; its 35 points are shown as *black dots*. The *red ones* are the corresponding loop-closure points; $n_c = 86 - 35 = 51$. For each frequency, we then have 51 estimable functions of carrier-wave ambiguities, i.e., 51 CD ambiguities. With regard to the selected spanning tree, 29 CDs are of order 4, 14 are of order 6, 3 are of order 8,

3 are of order 10, 1 is of order 12, and 1 is of order 14. In this scenario, the number of DDs is equal to 235: $n_{dd} = 235$. The maximum number of independent DDs is then equal to 47: $n_d^m = 47$. The information provided by these DDs does not reach that of the CDs. More precisely, these DDs define the CD-ambiguity vector up to a vector of K ; see Fig. 6; K is of dimension $n_c - n_d^m$, i.e., here 4. Note that no loop of order 4 includes points (4, 18), (10, 7), and (10, 18). For further details, see text

To illustrate our analysis in a more realistic manner, let us consider the scenario defined in Table 3.2 of de Jonge (1998): Scenario 3; see Fig. 7. In that case, $n_c = 51$ and $n_d^m = 47$. The dimension of K is then equal to 4; \mathbb{L} then includes four integers; more precisely, $\mathbb{L} = \{31, 32, 33, 40\}$. The basis vectors of K , the vectors f_{31} , f_{32} , f_{33} , and f_{40} , are then explicitly determined via Eq. (77). Clearly, we then have four K -ambiguities: $N_{v;31}^{[K]}$, $N_{v;32}^{[K]}$, $N_{v;33}^{[K]}$, and $N_{v;40}^{[K]}$.

10.3 Differenced methods

When the network master is not interested in the pseudo-clock biases, the methods based on the DD- or CD-differenced observational equations can be implemented. The position of some receivers (for example) can thus be refined.

When $n_d^m = n_c$, the sets of DD and CD ambiguities thus obtained are equivalent; see Eq. (75). The information provided by the CD-differenced observational data is then the same as that provided by the DD-differenced observational data.

When $n_d^m < n_c$, this is no longer the case: the information provided by the CD-differenced observational data is better than that provided by the DD-differenced observational data. The CD-differenced approach is then preferable; see for example Fig. 7.

10.4 Undifferenced methods

As specified in Sect. 9, the satellite pseudo-clock variables are of interest for precise point positioning. The methods based on the canonical undifferenced equations are then strongly recommended; see Sect. 7, Remark 7.1 and Sect. 8. This pointed out, if one wants to refer, somehow, to the DD approach, one may proceed in two other ways. We now describe the corresponding algebraic approaches.

10.4.1 DD parametrization

In the general case where n_d^m is strictly less than n_c , the canonical phase equation (51) can be parametrized in terms

of DD and K -ambiguities. In Sect. 8, the entries of the global variable v are then the DD ambiguities $N_{v;q}^{[dd]}$ and the K -ambiguities $N_{v;\ell}^{[K]}$. The matrix B of Eq. (55) is then built on the grounds of Eq. (78), i.e.,

$$N_v^{[cd]} = (D_H^{cd})^{-1} N_v^{[dd]} + \sum_{\ell \in \mathbb{L}} N_{v;\ell}^{[K]} f_\ell \quad (n_d^m < n_c)$$

In the case where n_d^m is equal to n_c , this equation reduces to the relation [see Eq. (75)]

$$N_v^{[cd]} = (D^{cd})^{-1} N_v^{[dd]} \quad (n_d^m = n_c)$$

The canonical phase equations are then parametrized in terms of DD ambiguities only.

10.4.2 DD initialization

In the usual DD approach, one works with the DD-differenced observational equations. Once the DD ambiguities have been fixed and validated, the CD ambiguity function $(D_H^{cd})^{-1} N_v^{[dd]}$ or $(D^{cd})^{-1} N_v^{[dd]}$ is also fixed and validated.

In the general case where n_d^m is strictly less than n_c , the K -ambiguities then remain to be fixed. For each frequency v , the matrix of the operator B of Eq. (55) then includes $n_c - n_d^m$ columns, one for each $\ell \in \mathbb{L}$. Up to the factor λ_v , the entries of the CD phase part of the ℓ th column are the components of the vector f_ℓ defined in Eq. (77). For n_v carrier waves, the dimension of the ambiguity problem to be solved is then reduced to $n_v(n_c - n_d^m)$; see Fig. 7 and Sect. 8.

In the special case where n_d^m is equal to n_c , no other ambiguity is to be fixed. The satellite pseudo-clock biases are then immediately obtained.

11 Concluding comments

The first objective of this paper was to show that the algebraic structures of some parametrized GNSS equations share a common feature with those of the phase calibration problems in Fourier synthesis (This Fourier synthesis technique is used in astronomy for imaging brightness sources at high resolution; see Lannes 2005). In fact, as specified in Remark 3.1, phase calibration problems are embedded in the processing of GNSS signals. The related key structure is that of Eqs. (21) and (25). In both cases, the phase data are defined on the edges of an observational graph, modulo λ or 2π ; see Eq. (23). Furthermore, these data are blurred by phase biases defined on the vertices of the graph. The algebraic graph theory and the algebraic number theory are thereby basically involved in the statement and the solution of the corresponding problems; see Sects. 4, 5, 7, and 8.

In astronomy, to get rid of the phase biases, ‘closure operations’ are performed along the loops of the observational graph (strictly speaking along the edges of the cycles defined

by an arbitrary spanning tree of this graph). This is why the corresponding Fourier techniques have led to the concept of phase closure imaging (PCI). Similar operations were introduced in GNSS. For example, double differencing consists in performing closure operations along loops including four edges.

More generally, the estimable functions of carrier-phase ambiguities (for example) are ‘closure-delay’ (CD) ambiguities associated with loops including an even number of edges; see Sect. 5 and Fig. 2. Clearly, this GNSS terminology derives from that used in PCI. In the related approach, referred to as Approach 1, Eq. (21) is regarded as an algebraic decomposition of a function taking its values on the edges of the GNSS graph; see Property 2.

The corresponding GNSS analysis had been performed, independently, via the S -basis technique developed by Baarda (1973), Teunissen (1984), and de Jonge (1998). In that approach, referred to as Approach 2 (Sect. 6), the idea is then to correct for the rank defect of Eq. (21). As illustrated in Fig. 3, the edge-delay space, as it is defined in Sect. 4.3, can be canonically decomposed into two components: the bias-delay space and the CD space. The corresponding property (Property 2) did not explicitly appear in Approach 2. In fact, as revealed by the analysis developed in Sects. 5 and 6, this property was simply hidden. The equivalence of Approaches 1 and 2 is therefore complete (However note that in Approach 1, the closure operator is built in a simpler manner; see Remark 6.1.). From an epistemological point of view, this equivalence had to be established. This is now done.

As already pointed out, the notion of closure delay can be regarded as a generalization of that of double difference (DD). The reader might therefore be surprised that CD ambiguities appear in the canonical undifferenced equation (51); and likewise for the canonical SD equations; see Remark 7.1. In fact, this is a simple consequence of Property 2.

A related application, which corresponded to the second objective of this paper, concerned the definition of the clock information to be broadcasted to the network users for their precise point positioning (PPP); see Sect. 9. It was then shown that this positioning can be achieved by simply having access to some satellite pseudo-clock biases. For simplicity, the study was restricted to relatively small networks. For example, as far as the phase is concerned, these biases then include five components (see Sect. 7): a frequency-dependent satellite-clock error, a tropospheric satellite delay, an ionospheric satellite delay, an initial satellite phase, and an integer satellite ambiguity. The form of the PPP equations to be solved by the network user is then similar to that of the traditional PPP equations; see Eqs. (62), (63) and Remark 9.1.

As soon as the CD ambiguities are fixed and validated (an operation which can be performed in real time via appropriate decorrelation techniques; see Sect. 8), estimates of these float biases can be immediately obtained. No other ambi-

guity is then to be fixed. The satellite pseudo-clock biases can therefore be obtained in real time. Note that this is not the case for the satellite-clock biases (see, e.g., Bertiger et al. 2010; Dow et al. 2009). In fact, extracting the satellite-clock biases and the satellite ambiguities from the satellite pseudo-clock biases requires some additional information or constraint, such as ‘statistical bounds’ on the clock biases (for example).

The third objective of this paper was to make the link between the CD approach and the GNSS methods based on the notion of double difference. As shown in Sect. 10, the information provided by a maximum set of independent DDs may not reach that of a complete set of CDs; see for example Fig. 7. To get the most from the GNSS data, it is therefore preferable to work with the canonical undifferenced equations, or with the CD-differenced equations if one is not interested in the pseudo-clock biases; see Sect. 10.3. This was already known in the GNSS community (de Jonge 1998), but the corresponding defect of the DD approach had not been yet completely analyzed. From this point of view, one of the main results of our contribution concerns the DD–CD relationship (Sect. 10.2).

The maximum number of independent DDs, n_d^m , is less than or equal to n_c , the number of CDs; see Eq. (68). In the special case of the scenarios for which n_d^m is equal to n_c , there exists a unimodular matrix that transforms the DD (column) vector into an equivalent CD vector. Once the DD ambiguities have been fixed and validated, the corresponding constraints can therefore be imposed in the undifferenced equations in an algebraic manner; see Sect. 10.4. To get the PPP information, it is therefore no longer required to introduce DD-ambiguity constraint equations as this is done (for instance) in the approach developed by Ge et al. (2005, 2006).

In the general case where n_d^m is strictly less than n_c , the DD ambiguity vector defines the CD ambiguity vector up to a vector in the null space of the DD operator restricted to the CD space; see Fig. 6. As clarified via Properties 3, 4, and Appendix 1, appropriate integer-programming algorithms allow the corresponding integer lattice to be completely identified. The DD ambiguities can thereby be still used as input data in the canonical phase equations. The integer-ambiguity problem that remains to be solved is then of smaller size; see Sect. 10.4.2.

The founding algorithm presented in Appendix 1 can also be used to find a maximum set of independent DDs. This is another interesting spin-off of our contribution; for instance, see Example 3 in Appendix A1.3, and compare with Saalfeld (1999).

Acknowledgments The second author is the recipient of an Australian Research Council Federation Fellowship (project number FF0883188). This support is gratefully acknowledged.

Open Access This article is distributed under the terms of the Creative Commons Attribution Noncommercial License which permits any noncommercial use, distribution, and reproduction in any medium, provided the original author(s) and source are credited.

Appendix 1: On some integer matrices

In this appendix, we consider a particular class \mathcal{M} of integer matrices. An $[m \times n]$ matrix $[A]$ lies in \mathcal{M} if its entries (i.e., its matrix elements $a_{i,j}$) are equal to ± 1 or 0. For the objectives of the paper, we also introduce two subclasses of \mathcal{M} : \mathcal{M}_1 and \mathcal{M}_2 (Appendices A1.1 and A1.2, respectively). A related founding algorithm is then presented (Appendix A1.3). Appendix A1.4 is finally devoted to the study of an important special case.

A1.1. Class \mathcal{M}_1

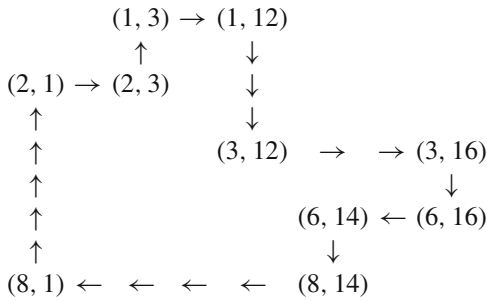
Let $[A]$ be some matrix in \mathcal{M} . Now, suppose that for some i, i' and j , we have either $a_{i,j} = \pm 1$ with $a_{i',j} = \mp 1$ (Case 1), or $a_{i,j} = \pm 1$ with $a_{i',j} = \pm 1$ (Case 2). We then denote by $[S]$ the unimodular matrix that reduces $a_{i',j}$ to 0. In Case 1, $[S]$ is the unimodular matrix $[S_+]$ that adds the entries of line i to the corresponding entries of line i' . In Case 2, $[S]$ is the unimodular matrix $[S_-]$ that subtracts the entries of line i from the corresponding entries of line i' . Note that when $a_{i,j} = -a_{i',j}$ for all j , $[S_+]$ sets to 0 all the elements of line i' . Likewise, when $a_{i,j} = a_{i',j}$ for all j , $[S_-]$ annihilates line i' .

By definition, \mathcal{M}_1 is the class of matrices $[A]$ for which the following property holds: *under successive operations with type $[S]$, $[A]$ remains in \mathcal{M} .*

Example 1 Let \mathcal{V} be a set including n vertices: $\mathcal{V} := \{v_1, v_2, \dots, v_n\}$

Design a matrix $[A]$ of \mathcal{M} as follows. For each line, choose two distinct vertices of \mathcal{V} : v_p and v_q . Then set $a_{i,p} = 1$, $a_{i,q} = -1$, and $a_{i,j} = 0$ for $j \neq p$ and $j \neq q$. Each line of $[A]$ is thus associated with a directed edge (v_p, v_q) of some graph. When two distinct edges (v_p, v_q) and $(v_{p'}, v_{q'})$ share only one vertex, the action of either S_+ or S_- amounts to associating another edge with line i' . As a result, $[A]$ lies in \mathcal{M}_1 . This shows, in particular, that the matrices $[B]$ and $[Q^{st}B]$ of Sect. 6 lie in \mathcal{M}_1 .

Example 2 Let us consider some scenario grid G ; see, for instance, that defined in Fig. 7. A rectangle whose vertices are four points of G forms a loop of order 4. As specified in Sect. 10.1, such loops are involved in the definition of double differences. More generally, one may design loops of order $2m$, with only two points per line or per column, in an alternate manner. For example, in Fig. 7, the following loop is of order 10:



We now design a matrix $[A]$ of \mathcal{M} as follows. Each line of $[A]$ is associated with some loop of even order. Along that loop, the value 1 or -1 is allocated to the successive points of G , in an alternate manner. The points of G being ordered line by line, we thus form the matrix of a ‘multi-difference operator’ which annihilates the vertex biases. Such a matrix lies in \mathcal{M}_1 . Indeed, when two loops of G with order $2m$ and $2m'$ share n points, the action of either $[S_+]$ or $[S_-]$ amounts to associating another loop with line i' . That loop is of order $2(m + m' - n)$. For example, two loops of order four sharing only one point thus yield a loop of order six. It is also clear that if some column(s) of such a matrix is (are) removed, the matrix thus obtained also lies in \mathcal{M}_1 . It then follows, in particular, that the matrices $[D]$, $[D^{cd}]$ and $[D_H^{cd}]$ introduced in Sects. 10.1 and 10.2 lie in \mathcal{M}_1 .

A1.2. Class \mathcal{M}_2

For each $i \leq m$, we denote by j_i the smallest index j for which $a_{i,j}$ is equal to ± 1 . If $a_{i,j}$ is equal to 0 for all $j \leq n$, we set $j_i = n + 1$.

Matrix $[A]$ lies in \mathcal{M}_2 if $[A]$ satisfies the following two conditions:

- (i) When $j_i \leq n$, then $a_{i,j_i} = 1$.
- (ii) When $j_i < n$, then j_{i+1} is strictly greater than j_i .
When $j_i = n$, then $j_{i'} = n + 1$ for $i' > i$.

Examples of such matrices are to be found in Appendix A1.3.

A1.3. Founding algorithm

Property For any $[A]$ in \mathcal{M}_1 , there exists a unimodular matrix $[Z]$ such that $[Z][A]$ lies in \mathcal{M}_2 .

The following algorithm, which yields $[Z][A]$ and $[Z]$ explicitly, can be regarded as a proof of this property.

Step 0 Initialization

Set $i_0 = 1$, $j_0 = 1$, and $[Z] = [I_m]$ where $[I_m]$ is the $[m \times m]$ identity matrix.

Step 1 Define the column j_0 to be processed

While $a_{i,j_0} = 0$ for all $i \geq i_0$ with $j_0 < n$, set $j_0 \stackrel{\text{set}}{=} j_0 + 1$. If $a_{i,j_0} = 0$ for all $i \geq i_0$ with $j_0 = n$, terminate the process.

Step 2 If need be, perform some line permutation

Denote by i_1 the smallest integer $i \geq i_0$ for which a_{i_1,j_0} is equal to ± 1 . When i_1 is not equal to i_0 , permute lines i_0 and i_1 . The corresponding operation can be written in the form $[A] \stackrel{\text{set}}{=} [P][A]$ where $[P]$ is a permutation matrix, a trivial unimodular matrix. Update $[Z] : [Z] \stackrel{\text{set}}{=} [P][Z]$.

Step 3 If need be, set $a_{i_0,j_0} = 1$

If $a_{i_0,j_0} = -1$, set $[A] \stackrel{\text{set}}{=} [S][A]$ and $[Z] \stackrel{\text{set}}{=} [S][Z]$, where $[S]$ is the trivial unimodular matrix that changes the sign of the entries of line i_0 .

Step 4 If need be, set to 0 the lower part of column j_0

For all $i > i_0$, if $a_{i,j_0} = \pm 1$, then combine lines i and i_0 so that $a_{i,j_0} = 0$. In other terms, set $[A] \stackrel{\text{set}}{=} [S_{\mp}][A]$, and update $[Z]$ consequently: $[Z] \stackrel{\text{set}}{=} [S_{\mp}][Z]$.

Step 5 Termination test

If $i_0 = m$ or $j_0 = n$, terminate the process. Otherwise, set $i_0 \stackrel{\text{set}}{=} i_0 + 1$, $j_0 \stackrel{\text{set}}{=} j_0 + 1$, and go to Step 1.

Example 3 Consider the matrix $[D]$ of the operator that provides all the DDs of Scenario 2 (see Sect. 10.1):

$$\begin{bmatrix}
 1 & 0 & -1 & -1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
 1 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 1 & 0 & 0 \\
 0 & 0 & 0 & 1 & -1 & 0 & 0 & 0 & 0 & -1 & 1 & 0 & 0 & 0 \\
 0 & 0 & 0 & 1 & 0 & -1 & 0 & 0 & 0 & -1 & 0 & 1 & 0 & 0 \\
 0 & 0 & 0 & 0 & 1 & -1 & 0 & 0 & -1 & 1 & 0 & 0 & 0 & 0 \\
 0 & 0 & 0 & 0 & 1 & -1 & 0 & 0 & 0 & 0 & -1 & 1 & 0 & 0 \\
 0 & 0 & 0 & 0 & 0 & 0 & 1 & -1 & 0 & 0 & 0 & 0 & -1 & 1 \\
 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & -1 & 0 & -1 & 1 & 0 & 0
 \end{bmatrix}$$

By construction (see Example 2), this matrix lies in \mathcal{M}_1 . The matrix $[Z][D]$ provided by the founding algorithm is then the following:

$$\begin{bmatrix}
 1 & 0 & -1 & -1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
 0 & 0 & 0 & 1 & 0 & -1 & 0 & 0 & 0 & -1 & 0 & 1 & 0 & 0 \\
 0 & 0 & 0 & 0 & 1 & -1 & 0 & 0 & 0 & 0 & -1 & 1 & 0 & 0 \\
 0 & 0 & 0 & 0 & 0 & 0 & 1 & -1 & 0 & 0 & 0 & 0 & -1 & 1 \\
 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & -1 & 0 & -1 & 1 & 0 \\
 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0
 \end{bmatrix}$$

Clearly, the rank of this matrix (the dimension of its range), which is also that of $[D]$, is equal to 5. The rank of its transpose is also equal to 5. More specifically, the first five lines, which are linearly independent, correspond to the maximum set of independent DDs given in Sect. 10.2. In fact, that set was obtained by applying to the list of all the DD loops, the successive permutations defined at Step 2 of the algorithm. Therefore, there exists some unimodular matrix $[Z]$, built without any permutation, which transforms the DD matrix of this maximum set into the matrix formed by the first five lines of the previous \mathcal{M}_2 matrix (The DD matrix of this maximum set is displayed in Sect. 10.1).

A1.4. Unimodular matrices of \mathcal{M}_1

Let us consider the special case where $[A]$ is an invertible $[n \times n]$ -matrix of \mathcal{M}_1 . Then there exists some unimodular matrix $[Z]$ such that $[Z][A]$ is an upper-triangular matrix $[U]$ with rank n and determinant 1; see Appendix A1.3. The determinant of A is then the same as that of $[Z]$: ± 1 ; $[A]$ is therefore unimodular; its inverse is then given by the relation

$$[A]^{-1} = [U]^{-1}[Z]$$

Indeed, $([U]^{-1}[Z])[A] = [U]^{-1}([Z][A]) = [U]^{-1}[U] = [I_n]$. In this case, the j th column of $[A]^{-1}$ is therefore obtained from the j th column of $[Z]$ by back-substitution.

Example 4 Consider the matrix $[A] \equiv [Q^{st}][B]$ introduced in Sect. 6 for Scenario 1 (see Remark 4.3):

$$[A] = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & -1 & 0 \\ 1 & 0 & 0 & 0 & 0 & -1 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & -1 & 0 & 0 \\ 0 & 0 & 1 & -1 & 0 & 0 \end{bmatrix}$$

Then

$$[U] = \begin{bmatrix} \mathbf{1} & 0 & 0 & 0 & 0 & 0 \\ 0 & \mathbf{1} & 0 & 0 & 0 & 0 \\ 0 & 0 & \mathbf{1} & -1 & 0 & 0 \\ 0 & 0 & 0 & \mathbf{1} & 0 & 0 \\ 0 & 0 & 0 & 0 & \mathbf{1} & 0 \\ 0 & 0 & 0 & 0 & 0 & \mathbf{1} \end{bmatrix}$$

and

$$[Z] = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & -1 & 0 \\ 1 & -1 & 0 & 0 & 0 & 0 \\ 1 & 0 & -1 & 0 & 0 & 0 \end{bmatrix}$$

Hence,

$$[A]^{-1} = [U]^{-1}[Z] = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & -1 & 1 \\ 0 & 0 & 0 & 1 & -1 & 0 \\ 1 & -1 & 0 & 0 & 0 & 0 \\ 1 & 0 & -1 & 0 & 0 & 0 \end{bmatrix}$$

Example 5 Consider the matrix $[A] \equiv [D_H^{cd}]$ introduced in Sect. 10.2:

$$[A] = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & -1 & 0 & 0 & 0 \\ -1 & 0 & 1 & 0 & 0 \end{bmatrix}$$

In that case, $[U]$ is the identity matrix, and

$$[Z] = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 & 0 \\ 1 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \end{bmatrix}$$

It then follows that $[A]^{-1}$ is equal to $[Z]$.

References

Baarda HA (1973) S-transformations and criterion matrices. Publications in Geodesy 1. Netherlands Geodetic Commission, Delft. 5
 Bertiger W, Desai SD, Haines B, Harvey N, Moore AW, Owen S, Weiss JP (2010) Single receiver phase ambiguity resolution with GPS data. *J Geod* 84:327–337
 Biggs N (1996) Algebraic graph theory, 2nd edn. Cambridge University Press, Cambridge
 Björck A (1996) Numerical methods for least-squares problems. SIAM
 de Jonge PJ (1998) A processing strategy for the application of the GPS in networks. PhD dissertation 46. Netherlands Geodetic Commission, Delft
 Dow JM, Neilan RE, Rizos C (2009) The International GNSS Service in a changing landscape of global navigation satellite systems. *J Geod* 83:191–198
 Ge M, Gendt G, Dick G, Zhang FP (2005) Improving carrier-phase ambiguity resolution in global GPS networks solutions. *J Geod* 79:103–110
 Ge M, Gendt G, Dick G, Zhang FP, Rothacher M (2006) A new data processing strategy for huge GNSS global networks. *J Geod* 80:199–203
 Ge M, Gendt G, Rothacher M, Shi C, Lui J (2008) Resolution of GPS carrier-phase ambiguities in precise point positioning (PPP) with daily observations. *J Geod* 82:389–399
 Lannes A (2005) A global analysis of the phase calibration operation. *J Opt Soc Am A* 22:697–707
 Lannes A, Gratton S (2008) QR implementation of GNSS centralized approaches. *J GPS* 7:133–147
 Lannes A, Gratton S (2009) GNSS networks in algebraic graph theory. *J GPS* 8:53–75
 Saalfeld A (1999) Generating basis sets of double differences. *J Geod* 73:291–297
 Teunissen PJG (1984) Generalized inverses, adjustment, the datum problem and S-transformations. In: Grafarend E, Sanso F (eds) Optimization of geodetic networks. Springer, Berlin, pp 11–55
 Teunissen PJG (1995) The least-squares ambiguity decorrelation adjustment: a method for fast GPS integer ambiguity estimation. *J Geod* 70:65–82
 Teunissen PJG, Kleusberg A (1998) GPS for geodesy, 2nd edn. Springer, Berlin
 Verhagen S, Teunissen PJG (2006) New global satellite system ambiguity resolution methods compared to existing approaches. *J Guid Control Dyn* 29:891–991
 Zumbege JF, Heflin MB, Jefferson DC, Watkins MM, Webb FH (1997) Precise point positioning for the efficient and robust analysis of GPS data from large networks. *J Geophys Res* 102:5005–5017