

On the Problem of Double-Filtering in PPP-RTK

Khodabandeh, A.; Teunissen, P. J.G.; Psychas, D.

DOI

[10.3390/s23010229](https://doi.org/10.3390/s23010229)

Publication date

2023

Document Version

Final published version

Published in

Sensors

Citation (APA)

Khodabandeh, A., Teunissen, P. J. G., & Psychas, D. (2023). On the Problem of Double-Filtering in PPP-RTK. *Sensors*, 23(1), Article 229. <https://doi.org/10.3390/s23010229>

Important note

To cite this publication, please use the final published version (if applicable).
Please check the document version above.

Copyright

Other than for strictly personal use, it is not permitted to download, forward or distribute the text or part of it, without the consent of the author(s) and/or copyright holder(s), unless the work is under an open content license such as Creative Commons.

Takedown policy

Please contact us and provide details if you believe this document breaches copyrights.
We will remove access to the work immediately and investigate your claim.

Article

On the Problem of Double-Filtering in PPP-RTK

A. Khodabandeh ^{1,*} , P. J. G. Teunissen ^{1,2,3}  and D. Psychas ⁴ ¹ Department of Infrastructure Engineering, The University of Melbourne, Melbourne 3010, Australia² Department of Geoscience and Remote Sensing, Delft University of Technology, 2628 CN Delft, The Netherlands³ GNSS Research Centre, Curtin University, Perth 6845, Australia⁴ European Space Agency (ESA/ESTEC), 2200 AG Noordwijk, The Netherlands

* Correspondence: akhodabandeh@unimelb.edu.au

Abstract: To obtain single-receiver Global Navigation Satellite System (GNSS) parameter solutions, the PPP-RTK user-filter combines measurements with time-correlated corrections that are separately computed by the filter of an external provider. The consequence of exercising such double-filtering is that the Kalman filter's standard assumption of having uncorrelated measurements in time becomes violated. This leads the user-filter to lose its 'minimum variance' property, thereby delivering imprecise parameter solutions. The solutions' precision-loss becomes more pronounced when one experiences an increase in the correction latency, i.e., the delay in time after the corrections are estimated and the time they are applied to the user measurements. In this contribution, we propose a new multi-epoch formulation for the PPP-RTK user-filter upon which both the uncertainty and the temporal correlation of the corrections are incorporated. By a proper augmentation of the user-filter state-vector, the corrections are jointly measurement-updated with the user parameter solutions. Supported by numerical results, the proposed formulation is shown to outperform its commonly used counterpart in the minimum-variance sense.

Keywords: global navigation satellite system (GNSS); integer ambiguity resolution enabled precise point positioning (PPP-RTK); Kalman filter; double-filtering; time-correlated corrections



Citation: Khodabandeh, A.; Teunissen, P.J.G.; Psychas D. On the Problem of Double-Filtering in PPP-RTK. *Sensors* **2023**, *23*, 229. <https://doi.org/10.3390/s23010229>

Academic Editor: Jari Nurmi

Received: 18 November 2022

Revised: 18 December 2022

Accepted: 19 December 2022

Published: 26 December 2022



Copyright: © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Integer carrier-phase ambiguity resolution-enabled precise point positioning, PPP-RTK, has the potential to enormously benefit from the state-space packaging of the positioning corrections to reduce their transmission rate, i.e., the frequency with which the corrections are to be provided to single-receiver global navigation satellite system (GNSS) users, see, e.g., [1–5]. However, a reduction in the transmission rate comes at the cost of delivering time-delayed corrections. Consequently, the user is required to time-predict the corrections so as to bridge the gap between the corrections' generation time and the user positioning time.

Therefore, next to the intrinsic uncertainty brought by the randomness of GNSS measurements, PPP-RTK corrections also inherit extra uncertainty that is associated with their time-prediction [6]. Should the characteristics of such correctional uncertainty, e.g., in terms of corrections' (co)variance matrices, be made available, the PPP-RTK user would then be in a position to employ rigorous estimation methods so as to achieve the *most-precise* (minimum-variance) parameter solutions [7]. In practice, however, a proper quality description of the corrections is often not provided to the user [8]. This is because the state-space representation of the corrections is aimed at minimizing the amount of information required to be made available. In the absence of such information, the approach commonly taken is to assume that the corrections are precise enough so that they can be treated as *non-random* [9]. The consequence of this practice is that the user estimation method becomes *suboptimal* in the sense that it loses its minimum-variance property and fails to provide the correct quality description of parameter solutions [10].

In our earlier contributions [7–9], we showed how the PPP-RTK user can limit the sub-optimality level of their parameter solutions when dealing with a ‘single epoch’ of data. Such single-epoch solutions do benefit from either closed-form expressions of the corrections’ variance matrix or their approximate versions. The topic of the present contribution concerns the scenario where the user aims to estimate their parameters using *multiple epochs* of data in a near real-time manner. The well-known candidate for the real-time computation of such multi-epoch solutions is the Kalman filter [11,12]. In this contribution, we show why existing formulations of the user’s Kalman filter are *misspecified*, and propose an alternative formulation that can largely avoid the precision loss of the user filtered solutions.

The structure of the paper is organized as follows. The problem of double-filtering in PPP-RTK is briefly reviewed in Section 2. The dependence of the user-filter on an external provider-filter is characterized, identifying the factors that lead to a misspecified stochastic model of the user-filter. In Section 3, we discuss potential choices with which the user can weight their time-correlated, ‘corrected’ observation vectors. Simulated examples are given to provide numerical insight into the consequences of such choices. Section 4 is devoted to the new formulation of the user-filter. The recursive structure of the filter is presented and the approximation on which the filter’s optimality is based is highlighted. To numerically demonstrate the superiority of the filter over existing formulations, a single-station PPP-RTK setup [7,13] is employed in Section 5. It is thereby shown how the filter responds to rather high correction *latency*, i.e., the delay in time after the corrections are generated and the time they are applied to the user data. Finally, summary and concluding remarks are provided in Section 6.

The following notation will be used in this paper. The underscore symbol indicates the ‘randomness’ of quantities. Thus, \underline{x} is random, while x is not. The hat $\hat{\cdot}$ and check $\check{\cdot}$ symbols indicate the solutions of unknown parameters. Thus, \hat{x} (or \check{x}) is a solution of x . The subscript $t|\tau$ of $\hat{x}_{t|\tau}$ indicates that $\hat{x}_{t|\tau}$ is a solution of x_t which is obtained based on all the observations collected up to and including the time-instant (epoch) τ . The covariance operator is denoted by $\text{Cov}(\cdot, \cdot)$, while the capital Q is reserved for (co)variance matrices.

2. Optimal Provider-Filter vs. Misspecified User-Filter

We commence with the (linearized) observation equations of a single-receiver PPP-RTK user at epoch i

$$\underline{u}_i = B_i b_i + C_i c_i + \underline{n}_i, \quad (1)$$

where the user observation vector \underline{u}_i , together with the zero-mean random noise \underline{n}_i , is linked to the user’s unknown parameter vector b_i and the unknown correction vector c_i through the full-rank design matrices B_i and C_i . The observation vector \underline{u}_i may contain the GNSS carrier-phase and pseudorange (code) measurements, with b_i containing the position coordinates, carrier-phase ambiguities, receiver clock parameters, and instrumental biases. On the other hand, the correction vector c_i may contain estimable forms of satellite orbit and clock parameters, atmospheric parameters, and phase/code biases [14–16].

With the sole use of their measurements, the single-receiver user cannot *unbiasedly* determine both the unknown vectors b_i and c_i . In other words, the augmented design matrix $[B_i, C_i]$ is rank-defect, meaning that part of b_i (or of c_i) has to be held fixed as \mathcal{S} -basis so that (1) becomes solvable for *biased* versions of b_i and c_i [15]. This of course does not suit user positioning. To obtain b_i unbiasedly, the PPP-RTK user needs to receive an unbiased solution of the correction vector c_i from an external provider, e.g., a network of permanent GNSS stations [1].

2.1. Non-Random Corrections as a Basic Principle

For the sake of argument, let us first assume that the correction vector c_i is determined by the provider with *no* uncertainty and is made available to the user. Accordingly, the user would correct their observation Equation (1) as follows

$$\underline{u}_i - C_i c_i = B_i b_i + \underline{n}_i \quad (2)$$

Since matrix B_i is of full-column rank, ‘single-epoch’ solutions of the user parameters b_i can then be unbiasedly computed. Instead of sticking to one single-epoch of data, the user may improve the precision of their parameter solutions by incorporating the temporal behaviour of some involved parameters. For instance, the user phase ambiguities remain constant in time unless a cycle-slip occurs. The rather stable instrumental biases can be linked in time by a random-walk process [17], while a polynomial dynamic model may be employed to capture the motion of the user position [7]. Such constraints between the user parameter vectors b_h ($h = i, i + 1 \dots$) can be expressed by the following dynamic models [18]

$$\underline{o}_h^b = b_h - \Phi^b b_{h-1} + \underline{w}_h^b, \quad h = i + 1, i + 2 \dots \quad (3)$$

where the randomness of the zero-sampled pseudo-observation \underline{o}_h^b is characterized by the process noises \underline{w}_h^b . The transition matrix Φ^b links the user parameters between two successive epochs. Thus, $\Phi_{(j-i)}^b = \prod_{h=i+1}^j \Phi^b$ ($j > i$) links the parameters from epoch i to epoch j .

The constraints (3) serve as extra observations to increase the user model’s redundancy, which is, the number of user observations minus the number of the estimable parameters involved. An increase in the redundancy strengthens the user model, thus improving the user parameter solutions [19]. In the context of PPP-RTK, however, the user parameter solutions are to be computed in a near real-time manner, demanding *recursive* estimation methods. The Kalman-filter [11,12] is known to be an *optimal* estimation method to handle such recursive computation in a *minimum-variance* sense. This optimality property relies on a key assumption, however, namely that the corrected observation vectors $\underline{u}_h - C_h c_h$ ($h = i, i + 1, \dots$) must be *uncorrelated* in time, and uncorrelated with the process noises \underline{w}_h^b ($h = i + 1, i + 2, \dots$). Likewise, the process noises \underline{w}_h^b must be uncorrelated in time. Provided that the user collects their measurements independently over time, the observations \underline{u}_h (and therefore their corrected versions $\underline{u}_h - C_h c_h$) fulfill such an assumption. This serves as a basic principle of PPP-RTK that is commonly exercised in practice [8]. As will be shown below, however, such a key assumption is violated if the non-random correction vector c_h in $\underline{u}_h - C_h c_h$ is replaced by a ‘random’ correction solution that has been computed by the external provider filter. As a consequence, the underlying model of the PPP-RTK user-filter becomes misspecified, losing its minimum-variance optimality property [10].

2.2. Optimal Provider-Filter

The role of a PPP-RTK provider is to estimate the unknown correction vectors c_h ($h = i, i + 1, \dots$) and make the corresponding solutions available to the user. As with the user, the provider formulates their own Kalman filter with the observation equations (at epoch t)

$$\underline{y}_t = A_t c_t + \underline{e}_t, \quad t = 1, 2, \dots, \quad (4)$$

and the dynamic models

$$\underline{o}_t^c = c_t - \Phi^c c_{t-1} + \underline{w}_t^c, \quad t = 2, 3 \dots \quad (5)$$

where the provider observation vector \underline{y}_t , together with the zero-mean measurement noise \underline{e}_t , are linked to the unknown corrections vector c_t through the full-rank design matrix A_t . The transition matrix Φ^c links the correction vectors between two successive epochs with the zero-sampled pseudo-observation \underline{o}_t^c and time-uncorrelated process noises \underline{w}_t^c ($t = 2, 3 \dots$). If the observation vectors \underline{y}_t ($t = 1, 2, \dots$) are also time-uncorrelated, and uncorrelated with the process noises \underline{w}_t^c , the provider would then be in a position to run their optimal minimum-variance filter.

The three-step structure of the provider-filter is presented in the left-panel of Figure 1. The structure follows from an application of the least-squares principle to (4) and (5), see, e.g., [18]. At the initialization step (first epoch $t = 1$), the provider initializes their filter

by the least-squares solution $\hat{\mathbf{e}}_{1|1} = \mathbf{A}_1^+ \mathbf{y}_1$ with $\mathbf{A}_1^+ = (\mathbf{A}_1^T \mathbf{Q}_{y_1}^{-1} \mathbf{A}_1)^{-1} \mathbf{A}_1^T \mathbf{Q}_{y_1}^{-1}$, where \mathbf{Q}_{y_t} denotes the variance matrix of the observation vector \mathbf{y}_t ($t = 1, 2, \dots$). The application of the (co)variance propagation law also gives the solution's error-variance matrix as $\mathbf{Q}_{\hat{\mathbf{e}}_{1|1}} = (\mathbf{A}_1^T \mathbf{Q}_{y_1}^{-1} \mathbf{A}_1)^{-1}$. At the time-update (TU) step, the correction vector of the upcoming epoch can be time-predicted as $\hat{\mathbf{e}}_{t|t-1} = \Phi^c \hat{\mathbf{e}}_{t-1|t-1}$ ($t = 2, 3, \dots$), with the error-variance matrix $\mathbf{Q}_{\hat{\mathbf{e}}_{t|t-1}} = \Phi^c \mathbf{Q}_{\hat{\mathbf{e}}_{t-1|t-1}} \Phi^{cT} + \mathbf{Q}_{w_t^c}$, where $\mathbf{Q}_{w_t^c}$ is the variance matrix of w_t^c . Finally, at the measurement-update (MU) step, the filter makes use of the upcoming observation vector \mathbf{y}_t to recursively update the correction solution as $\hat{\mathbf{e}}_{t|t} = \hat{\mathbf{e}}_{t|t-1} + \mathbf{K}_t^c (\mathbf{y}_t - \mathbf{A}_t \hat{\mathbf{e}}_{t|t-1})$ with the error-variance matrix $\mathbf{Q}_{\hat{\mathbf{e}}_{t|t}} = (\mathbf{I} - \mathbf{K}_t^c \mathbf{A}_t) \mathbf{Q}_{\hat{\mathbf{e}}_{t|t-1}}$, in which the Kalman gain matrix is given by $\mathbf{K}_t^c = \mathbf{Q}_{\hat{\mathbf{e}}_{t|t-1}} \mathbf{A}_t^T (\mathbf{Q}_{y_t} + \mathbf{A}_t \mathbf{Q}_{\hat{\mathbf{e}}_{t|t-1}} \mathbf{A}_t^T)^{-1}$.

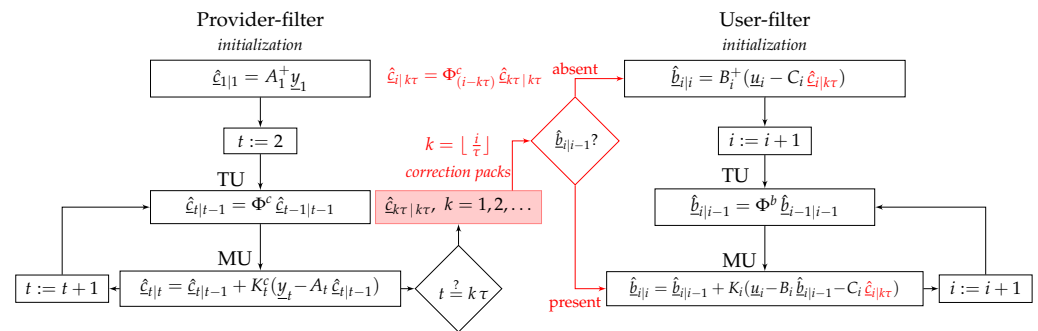


Figure 1. Double-filtering in PPP-RTK. **(Left):** the provider uses ‘time-uncorrelated’ observation vectors \mathbf{y}_t ($t = 0, 1, \dots$) to generate and disseminate ‘time-correlated’ correction packs $\hat{\mathbf{e}}_{k\tau|k\tau}$ ($k = 1, 2, \dots$) every τ seconds. **(Right):** the user picks the ‘most recent’ correction pack to feed their model at epoch i . Thus, $k = \lfloor i/\tau \rfloor$, where the operator $\lfloor x \rfloor$ returns the greatest integer less than or equal to x . If no previous user solution $\hat{\mathbf{e}}_{i|i-1}$ is present, the user initializes their filter with the corrected observation vector $\mathbf{u}_i - \mathbf{C}_i \hat{\mathbf{e}}_{i|k\tau}$. Otherwise, the correction pack is fed into the user MU step.

2.3. Misspecified User-Filter

In contrast to the provider-filter, the user-filter cannot stand on its own and requires the output of the provider-filter, that is, the correction solutions $\hat{\mathbf{e}}_{h|h}$ ($h = i, i+1, \dots$). Due to the intrinsic randomness of the provider observations, $\hat{\mathbf{e}}_{h|h}$ is accompanied with an amount of *uncertainty* characterized by the error-variance matrix $\mathbf{Q}_{\hat{\mathbf{e}}_{h|h}}$. With reference to (2), this implies that the random filtered solutions $\hat{\mathbf{e}}_{h|h}$ replace their non-random versions \mathbf{c}_h to form the user *corrected* observation vectors $\mathbf{u}_h - \mathbf{C}_h \hat{\mathbf{e}}_{h|h}$. As the observation vectors \mathbf{u}_h are time-uncorrelated, it is the correction solutions $\hat{\mathbf{e}}_{h|h}$ that dictate whether the Kalman filter’s key assumption holds. This is followed by applying the covariance propagation law to corrected observation vectors of any two distinct epochs j and i , that is

$$\text{Cov}(\mathbf{u}_j - \mathbf{C}_j \hat{\mathbf{e}}_{j|j}, \mathbf{u}_i - \mathbf{C}_i \hat{\mathbf{e}}_{i|i}) = \mathbf{C}_j \mathbf{Q}_{\hat{\mathbf{e}}_{j|j}, \hat{\mathbf{e}}_{i|i}} \mathbf{C}_i^T, \quad j \neq i \quad (6)$$

in which the covariance matrix $\mathbf{Q}_{\hat{\mathbf{e}}_{j|j}, \hat{\mathbf{e}}_{i|i}}$ between the correction solutions $\hat{\mathbf{e}}_{j|j}$ and $\hat{\mathbf{e}}_{i|i}$ is shown to read (Appendix A).

$$\mathbf{Q}_{\hat{\mathbf{e}}_{j|j}, \hat{\mathbf{e}}_{i|i}} = \left[\prod_{h=i+1}^j (\mathbf{I} - \mathbf{K}_h^c \mathbf{A}_h) \Phi^c \right] \mathbf{Q}_{\hat{\mathbf{e}}_{i|i}}, \quad j > i \quad (7)$$

The nonzero covariance matrix above indicates that the user-corrected observation vectors $\mathbf{u}_h - \mathbf{C}_h \hat{\mathbf{e}}_{h|h}$ ($h = i, i+1, \dots$) are indeed time-correlated, making the user-filter misspecified.

Although the PPP-RTK user-filter is *not* minimum-variance, and thus suboptimal, previous studies have demonstrated that the filter can still deliver successful ambiguity-resolved positioning solutions when the duration of the provider-filter initialization, i.e.,

the time-difference between the epoch i and the initial epoch $t = 1$, becomes sufficiently large (e.g., ~ 1 h), as can be seen in, e.g., [6,7,9]. In other words, the filtered solution $\hat{\mathbf{c}}_{i|i}$ can become sufficiently precise so as to neglect its uncertainty relative to that of the user observations, i.e., $Q_{\hat{\mathbf{c}}_{i|i}} \approx 0$. By making such an approximation, the covariance matrices in (6) and (7) become zero, meaning that the user-filter is expected to deliver minimum-variance solutions. While the duration of the provider-filter initialization can be ensured to be sufficiently long to make that approximation, the corrections *cannot* be *instantaneously* transferred to the user due to the limited data-transmission bandwidth [1]. This is all the more so as PPP-RTK is meant to take advantage of the efficient ‘state-space’ packaging of the corrections to reduce their transmission rate, i.e., the frequency with which the corrections are to be provided to the user. Consequently, PPP-RTK corrections ranging from orbits and clocks to phase biases are stored in, e.g., an Internet server, each having its own sampling period τ . In contrast to RTK for which the correction latency is less than 4 s [20] (i.e., $\tau \leq 4$), PPP and PPP-RTK state-space corrections are to be provided with higher time-delays. For instance, current PPP real-time service of IGS (<https://igs.org/rtts/>) (accessed on 18 December 2022) disseminates state-space corrections with a typical latency of 5–10 s, see, e.g., [21,22]. In the following, it is shown how such state-space packaging brings additional correctional uncertainty.

The red-box in Figure 1 indicates the ‘correction packs’ $\hat{\mathbf{c}}_{k\tau|k\tau}$ ($k = 1, 2, \dots$) that are generated and stored by the provider every τ s. As a result, the user gains access to the correction packs with a time-delay or ‘latency’, i.e., $k\tau \leq i$. The correction latency $i - k\tau$ ranges from 0 to $\tau - 1$ s. As shown in the right-panel of Figure 1, the user picks the ‘most recent’ correction pack and *time-predicts* the correction as $\hat{\mathbf{c}}_{i|k\tau} = \Phi_{(i-k\tau)}^c \hat{\mathbf{c}}_{k\tau|k\tau}$ to feed their model at epoch i . Thus, even if the approximation $Q_{\hat{\mathbf{c}}_{k\tau|k\tau}} \approx 0$ would be plausible, the uncertainty associated with the time-predicted correction $\hat{\mathbf{c}}_{i|k\tau}$ may not be negligible. This is indeed the case when the time-difference $i - k\tau$ is considered to be large. To see this, consider the error-variance matrix of $\hat{\mathbf{c}}_{i|k\tau}$ as (Appendix A)

$$Q_{\hat{\mathbf{c}}_{i|k\tau}} = \Phi_{(i-k\tau)}^c Q_{\hat{\mathbf{c}}_{k\tau|k\tau}} \Phi_{(i-k\tau)}^{cT} + \sum_{h=k\tau+1}^i \Phi_{(i-h)}^c Q_{w_h^c} \Phi_{(i-h)}^{cT} \quad (8)$$

While the first term in (8) may be considered negligible for a long duration of the provider-filter initialization, the second term increases as the latency or the time-difference $i - k\tau$ increases, leading to a misspecified user-filter. Note, for the sake of presentation, that we did not distinguish between each individual correction type (e.g., satellite orbits versus clocks) in Figure 1. We instead only show one common sampling period τ for all correction types. In practice however, each individual correction can, of course, have its own sampling period τ .

The three-step structure of the misspecified user-filter follows, in a way analogous to that of the provider-filter, be it that the role of the correction parameters \mathbf{c}_t is replaced by the user parameters \mathbf{b}_i , the design matrix \mathbf{A}_t by \mathbf{B}_i , and the observation vectors \mathbf{y}_t by the user corrected observation vectors $\mathbf{u}_i - \mathbf{C}_i \hat{\mathbf{c}}_{i|k\tau}$ (Figure 1, right-panel). In contrast to the provider who weights the observation vectors \mathbf{y}_t using the inverse of their variance matrix $\mathbf{Q}_{\mathbf{y}_t}$, the user may not have access to the variance matrix of the correction $\hat{\mathbf{c}}_{i|k\tau}$ to properly weight the corrected observation vectors $\mathbf{u}_i - \mathbf{C}_i \hat{\mathbf{c}}_{i|k\tau}$. Instead, let us assume that the user takes a given inverse-weight matrix $\bar{\mathbf{Q}}_{\mathbf{u}_i}$ to replace its provider-counterpart $\mathbf{Q}_{\mathbf{y}_i}$. As shown in [10], the misspecified user-filter would then report *incorrect* error-variance matrices as

$$\begin{aligned} \text{(initialization):} \quad \bar{\mathbf{Q}}_{\hat{\mathbf{b}}_{i|i}} &= (\mathbf{B}_i^T \bar{\mathbf{Q}}_{\mathbf{u}_i}^{-1} \mathbf{B}_i)^{-1}, \quad i := i + 1 \\ \text{(time-update):} \quad \bar{\mathbf{Q}}_{\hat{\mathbf{b}}_{i|i-1}} &= \Phi_{i-1}^b \bar{\mathbf{Q}}_{\hat{\mathbf{b}}_{i-1|i-1}} \Phi_{i-1}^{bT} + \mathbf{Q}_{w_i^b}, \\ \text{(measurement-update):} \quad \bar{\mathbf{Q}}_{\hat{\mathbf{b}}_{i|i}} &= (\mathbf{I} - \mathbf{K}_i \mathbf{B}_i) \bar{\mathbf{Q}}_{\hat{\mathbf{b}}_{i|i-1}} \end{aligned} \quad (9)$$

where the Kalman gain matrix is evaluated as $K_i = \bar{Q}_{\hat{b}_{|i-1}} B_i^T (\bar{Q}_{u_i} + B_i \bar{Q}_{\hat{b}_{|i-1}} B_i^T)^{-1}$. Here and in the following, the $\bar{\cdot}$ -symbol over the Capital Q is used to distinguish incorrect variance matrices. Thus, not only does the user-filter deliver suboptimal solutions, but it also fails to provide the correct quality description of the user parameter solution $\hat{b}_{|i}$. In the following, we discuss a choice of the inverse-weight matrix \bar{Q}_{u_i} which is often adopted in practice.

3. On the Choice of the Inverse-Weight Matrix \bar{Q}_{u_i}

In the previous section, it was shown that the main input of the user-filter, i.e., the corrected observation vectors $\underline{u}_h - C_h \hat{e}_{h|k\tau}$ ($h = i, i+1, \dots$), are time-correlated cf. (6), prohibiting the *recursive* computation of the *most precise* parameter solutions. By adopting the inverse-weight matrix \bar{Q}_{u_h} for $\underline{u}_h - C_h \hat{e}_{h|k\tau}$, however, the user can still recursively compute suboptimal parameter solutions using their misspecified filter (cf. Figure 1). Relying on the assumption that the filtered corrections $\hat{e}_{k\tau|k\tau}$ are sufficiently precise to make the approximation $Q_{\hat{e}_{k\tau|k\tau}} \approx 0$, one may choose \bar{Q}_{u_h} in a way to limit the sub-optimality level of the parameter solutions. For instance, if the uncertainty involved in the time-prediction of the *time-delayed* corrections $\hat{e}_{h|k\tau}$ can be ignored (i.e., the second term in (8) is neglected), the inverse-weight matrix \bar{Q}_{u_h} can then be set to the variance matrix of the user observations \underline{u}_h , that is

$$\text{Case 1 : } \bar{Q}_{u_h} := Q_{u_h}, \quad h = i, i+1, \dots \quad (10)$$

For this case, the user only takes the variance matrix of their own data, i.e., Q_{u_h} , for the weighting of the corrected data $\underline{u}_h - C_h \hat{e}_{h|k\tau}$. In other words, the external corrections $\hat{e}_{h|k\tau}$ are considered sufficiently precise to be treated as *non-random*, the scenario that is commonly exercised in practice [7,8]. This is because the corrections' error variance matrix $Q_{\hat{e}_{k\tau|k\tau}}$ is often not provided to the user. After all, the purpose of using state-space PPP-RTK corrections is to minimize the amount of information to be transmitted to the user [1]. The following example shows the consequence of this choice, i.e., Case 1 (10).

Example 1. To give primary numerical insight into the consequence of choosing (10), consider a single-receiver user with a known location who is tracking the L1/L2 dual-frequency code data of a pair of satellites to determine the corresponding single-differenced (SD) slant ionospheric delay over 100 epochs with 1 Hz measurement sampling-rate. The user is given a satellite clock offset- and rate-corrections every $\tau = 10$ s. Thus, the correction latency ranges from 0 to 9 s. In this simulation example, the filtered corrections $\hat{e}_{k\tau|k\tau}$ are assumed and simulated to be non-random. Thus, $Q_{\hat{e}_{k\tau|k\tau}} = 0$. However, the user still has to time-predict the satellite clock corrections $\hat{e}_{i|k\tau}$ at every epoch i to compute their filtered ionospheric solution $\hat{b}_{|i}$. The time-behaviour of the undifferenced ionospheric delays is modelled by a random-walk process with a standard-deviation of $1 \text{ mm}/\sqrt{s}$, whereas the undifferenced satellite clocks follow a constant-velocity dynamic model with a standard-deviation of $1 \text{ cm}/\sqrt{s^3}$, as can be seen in, e.g., [6]. The standard-deviation of the undifferenced code data is set to 20 cm.

To measure how the misspecified user filter under Case 1 performs, 1000 normally distributed samples of both the correction and user data are simulated over the fixed 100 epochs. The corresponding samples of the ionospheric estimation-error $b_h - \hat{b}_{h|h}$ ($h = 1, \dots, 100$), i.e., the difference between the true simulated ionospheric parameter and its filtered solution, are shown in the left-panel of Figure 2 (grey lines). The black solid lines indicate the associated 99.9% confidence-intervals, whereas the dashed lines represent the corresponding incorrect confidence-intervals which are reported by the user filter. As the number of user epochs increases, the dispersion of the estimation-error becomes smaller, showing that the precision of the user filtered solution improves. However, more than 100 epochs of data are required to have the absolute value of the 99.9% confidence-intervals smaller than 1 dm. On the contrary, the filter reports rather optimistic confidence-intervals. The considerable gap between the correct and incorrect confidence-intervals of the user parameter solutions

is due to the choice made in (10), i.e., both the variance and time-correlation of the time-predicted corrections are ignored.

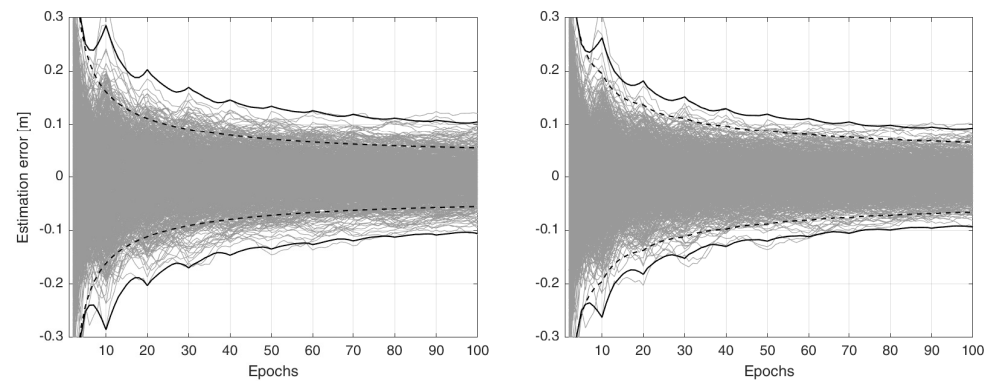


Figure 2. Time-series of the ionospheric estimation-error $b_h - \hat{b}_{h|h}$ ($h = 1, \dots, 100$) of Example 1 (grey lines). Given 1000 normally distributed samples, the misspecified user-filter computes filtered ionospheric solutions under two cases. **(Left):** Case 1 (10) in which the corrections are considered non-random. **(Right):** State-augmentation (12) in which the time-correlation of the corrections is modelled by an exponential auto-correlation function. The incorrect 99.9% confidence intervals (dashed lines), reported by the user-filter, are compared with their correct versions (black solid lines).

As the random corrections $\hat{c}_{i|k\tau}$ make the observation vectors $\underline{u}_i - C_i \hat{c}_{i|k\tau}$ time-correlated, one may employ the technique of first-order Markov state-augmentation to handle time-correlated measurements of the Kalman filter, as can be seen in, e.g., [23] (p. 180) or [24]. In that technique, the time-correlation of the corrections is ‘approximated’ by an exponential auto-correlation function [18], that is (compare with (7))

$$\bar{Q}_{\hat{c}_{j|j}, \hat{c}_{i|i}} := \bar{Q}_c e^{-\frac{1}{\alpha}|j-i|}, \quad \forall i, j \quad (11)$$

where the matrix \bar{Q}_c is to capture the uncertainty of the correction at every epoch. Next to the inverse-weight matrix \bar{Q}_{u_h} in (10), the state-augmentation technique incorporates the corrections’ variance and time-correlation via the matrices $\bar{Q}_{\hat{c}_{j|j}, \hat{c}_{i|i}}$ to weight both the uncorrected user observations \underline{u}_h and the time-predicted corrections $\hat{c}_{h|k\tau}$. The coefficient α governs the magnitude of the correlation between the observation vectors. The larger the coefficient α , the larger the time-correlation is assumed between the observation vectors. To run their filter in recursive form, the user would need to make the following modifications [18]

$$B_h \mapsto [B_h, C_h], \quad b_h \mapsto \begin{bmatrix} b_h \\ a_h \end{bmatrix}, \quad \Phi^b \mapsto \begin{bmatrix} \Phi^b & 0 \\ 0 & e^{-\frac{1}{\alpha}I} \end{bmatrix}, \quad \bar{Q}_{u_h} \mapsto \begin{bmatrix} \bar{Q}_{u_h} & 0 \\ 0 & \bar{Q}_c \end{bmatrix}, \quad \bar{Q}_{w_h^b} \mapsto \begin{bmatrix} \bar{Q}_{w_h^b} & 0 \\ 0 & \bar{Q}_c(1 - e^{-\frac{2}{\alpha}}) \end{bmatrix} \quad (12)$$

for epochs $h > i$, where the notation ‘ $A \mapsto B$ ’ means ‘replace A by B ’. Thus, the user state-vector b_h is augmented with the parameter vector a_h whose process noises’ time-correlation exponentially decays over time. For the user initial epoch $h = i$, the filter is initialized by the augmented state-vector $[\hat{b}_{i|i}^T, \hat{a}_{i|i}^T]^T$, with $\hat{b}_{i|i} = B_i^+(\underline{u}_i - C_i \hat{c}_{i|k\tau})$ and $\hat{a}_{i|i} = 0$. Thus, the initial user solution $\hat{b}_{i|i}$ is identical to that of Case 1 (10). For the upcoming epochs $h > i$, however, the filter aims to capture the randomness of the corrections by updating the solutions $\hat{b}_{h|h}$ over time.

To see the extent to which the state-augmentation technique can alleviate the effect of the time-correlated corrections, we evaluate the ionospheric estimation errors using the coefficients $\alpha = 50$ and $\bar{Q}_c = 0.02 \text{ m}^2$. These coefficients are empirically chosen so as to approximate the time-correlation of the corrections. The results, together with their correct and incorrect 99.9% confidence intervals, are depicted in the right-panel of Figure 2. As shown, the precision of the user filtered solution slightly improves, that is, the absolute value of its 99.9% confidence-intervals reaches 1 dm after 75 epochs. The gap between the correct and incorrect confidence-intervals also becomes smaller.

As shown in Example 1, treating the time-delayed PPP-RTK corrections as non-random leads to an incorrect and optimistic quality description of the user parameter solutions. Alternatively, the state-augmentation technique may be used to ‘approximate’ both the variance and time-correlation of the corrections via the exponential auto-correlation function (11). Such a technique does, however, *not* take advantage of the information contained in the corrections’ dynamic model (5), thereby discarding the structure of the error-variance matrix (8). We now consider another case that incorporates (8) into the user-filter. As stated before, the approximation $Q_{\hat{c}_{k\tau|k\tau}} \approx 0$ may hold by letting the duration of the provider-filter initialization be sufficiently long. The substitution of $Q_{\hat{c}_{k\tau|k\tau}} = 0$ into (8) gives an inverse-weight matrix $\bar{Q}_{\hat{c}_{i|k\tau}}$ for weighting the time-predicted corrections $\hat{c}_{i|k\tau}$. Accordingly, the user inverse-weight matrix \bar{Q}_{u_h} in (10) is modified as follows

$$\text{Case 2 : } \bar{Q}_{u_h} := Q_{u_h} + C_i \bar{Q}_{\hat{c}_{i|k\tau}} C_i^T, \quad h = i, i+1, \dots, \quad (13)$$

$$\text{with } \bar{Q}_{\hat{c}_{i|k\tau}} = \sum_{h=k\tau+1}^i \Phi_{(i-h)}^c Q_{w_h^c} \Phi_{(i-h)}^{cT}$$

Thus, the inverse-weight matrix $\bar{Q}_{\hat{c}_{i|k\tau}}$ only considers the second term in (8), i.e., the uncertainty due to the time-prediction of the corrections. Although the randomness of PPP-RTK corrections is taken into account under Case 2 (13), their nonzero time-correlation (7) is still dismissed. As with Case 1 (10), such a time-correlation dismissal is required to run the user-filter in its recursive form. In the next section, we show how Case 2 can be generalized so as to account for the corrections’ time-correlation (7), yet allowing the recursive estimation of user parameters.

4. User-Filter with Correctional Update

To date, the focus has been restricted to the choice of the user inverse-weight matrix \bar{Q}_{u_h} that can be made in Cases 1 and 2. Depending on how the user weights their corrected observation vectors $\underline{u}_h - \hat{c}_{h|k\tau}$ ($h = i, i+1, \dots$) through \bar{Q}_{u_h} , their misspecified filter can be recursively run in accordance with the right-panel of Figure 1. In both Cases 1 and 2, however, the time-predicted corrections $\hat{c}_{h|k\tau}$ do *not* benefit from the information contained in the user observations \underline{u}_h . In other words, the corrections are merely derived from the correction packs $\hat{c}_{k\tau|k\tau}$, that is $\hat{c}_{h|k\tau} = \Phi_{(h-k\tau)}^c \hat{c}_{k\tau|k\tau}$ over the epochs $h = i, \dots, (k+1)\tau - 1$.

As with the first-order Markov state-augmentation (12), the idea is to augment the user state-vector, with a difference, so that the correction parameter vector c_h now replaces the parameter vector a_h . Thus, instead of approximating the corrections’ time-correlation by an exponential auto-correlation function (11), the goal is now to directly incorporate the corrections’ dynamic model (5) into the user-filter. To this end, we again assume that the approximation $Q_{\hat{c}_{k\tau|k\tau}} \approx 0$ holds. The initialization of the user-filter is then executed through the following modifications

$$\text{Case 3 (initialization) : } b_i \mapsto \begin{bmatrix} b_i \\ c_i \end{bmatrix}, B_i \mapsto \begin{bmatrix} B_i & C_i \\ 0 & I \end{bmatrix}, (\underline{u}_i - C_i \hat{c}_{i|k\tau}) \mapsto \begin{bmatrix} \underline{u}_i \\ \hat{c}_{i|k\tau} \end{bmatrix}, \bar{Q}_{u_i} \mapsto \begin{bmatrix} Q_{u_i} & 0 \\ 0 & \bar{Q}_{\hat{c}_{i|k\tau}} \end{bmatrix} \quad (14)$$

where the inverse-weight matrix $\bar{Q}_{\hat{c}_{i|k\tau}}$ follows from that of (13). In contrast to Cases 1 and 2 in which the corrections are merely time-updated from $\hat{c}_{k\tau|k\tau}$, we now let the user-filter, next to the time-updating, also *measurement-update* the corrections $\hat{c}_{i|k\tau}$. We therefore employ the discriminating notation $\check{\cdot}$ (instead of \cdot) to denote the user-augmented parameter solutions $[\check{b}_{i|i}, \check{c}_{i|i}^T]^T$. During the period in which no newer correction pack is available, the user-filter recursively performs the time- and measurement-updates via the following settings

$$\text{Case 3 (TU-MU) : } (\underline{u}_h - C_h \hat{c}_{h|k\tau}) \mapsto \underline{u}_h, \bar{Q}_{u_h} \mapsto Q_{u_h}, B_h \mapsto [B_h, C_h],$$

$$\Phi^b \mapsto \begin{bmatrix} \Phi^b & 0 \\ 0 & \Phi^c \end{bmatrix}, Q_{w_h^b} \mapsto \begin{bmatrix} Q_{w_h^b} & 0 \\ 0 & Q_{w_h^c} \end{bmatrix}, \quad h = i+1, \dots, (k+1)\tau - 1 \quad (15)$$

At epoch $i = (k + 1)\tau$, when the new correction pack $\hat{\mathbf{c}}_{(k+1)\tau|(k+1)\tau}$ is made available by the provider, the new time-predicted corrections $\check{\mathbf{c}}_{i|(k+1)\tau}$ are to replace the existing user-filtered corrections $\check{\mathbf{c}}_{i|i}$. This follows from the approximation $Q_{\hat{\mathbf{c}}_{(k+1)\tau|(k+1)\tau}} \approx 0$ ($k = 1, 2, \dots$), namely that the new corrections $\hat{\mathbf{c}}_{i|(k+1)\tau}$ contain all the information contained in the user-filtered corrections $\check{\mathbf{c}}_{i|i}$. The structure of our proposed user-filter is presented in Figure 3. As shown, the user-filtered corrections $\check{\mathbf{c}}_{i|i}$ have to be initialized and replaced by $\hat{\mathbf{c}}_{i|k\tau}$ every time a newer correction pack $\hat{\mathbf{c}}_{k\tau|k\tau}$ is made available. Note that such a filter is still misspecified, thus delivering suboptimal solutions. However, its sub-optimality level is only dictated by the extent to which the error-variance matrix $Q_{\hat{\mathbf{c}}_{k\tau|k\tau}}$ is different from zero. This is in marked contrast to Cases 1 and 2, whose solutions' loss of precision is also driven by the correction latency $i - k\tau$. To compare the performance of the proposed filter with those of Cases 1 and 2, let us again consider the problem in Example 1.

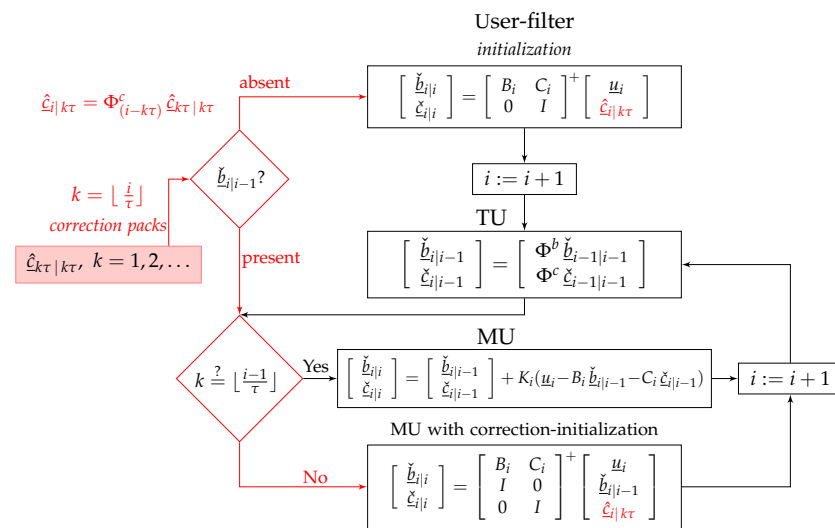


Figure 3. The proposed PPP-RTK user-filter with a correctional update. In the absence of user solution $\check{\mathbf{b}}_{i|i-1}$, the user picks the ‘most recent’ corrections $\hat{\mathbf{c}}_{i|k\tau}$ and initializes their filter via the augmented observation vector $[\mathbf{u}_i^T, \hat{\mathbf{c}}_{i|k\tau}^T]^T$. The filter does not only deliver the user parameter solution $\check{\mathbf{b}}_{i|i}$, but also recursively updates the corrections as $\check{\mathbf{c}}_{i|i}$ ($i := i + 1$). At every MU step, if a newer correction pack $\hat{\mathbf{c}}_{k\tau|k\tau}$ is made available by the provider (i.e., if $k > \lfloor \frac{i-1}{\tau} \rfloor$), then the user-filtered corrections $\check{\mathbf{c}}_{i|i}$ are initialized by $\hat{\mathbf{c}}_{i|k\tau}$.

Example 2 (Continuation of Example 1). Given the same 1000 normally distributed samples of both the correction and user data in Example 1, we evaluate the ionospheric estimation-errors $b_h - \hat{b}_h$ ($h = 1, \dots, 100$) by the user-filter using the settings om Case 2 (13) and our proposed settings as outlined in Figure 3. The corresponding results are presented in Figure 4. In comparison to the results in Figure 2, Case 2 outperforms Case 1, showing a similar performance to that of the state-augmentation technique (12), i.e., it requires at least 70 epochs to have solutions more precise than 1 dm (with 99.9% confidence). Similar to Case 1 and the state-augmentation technique, there is also a gap between the correct and incorrect confidence intervals. The proposed filter does, however, deliver the most precise results in the sense that the absolute value of their 99.9% confidence-intervals becomes smaller than 1 dm after 50 epochs. At the same time, there is no gap between the correct and reported confidence intervals. This is because the correction packs $\hat{\mathbf{c}}_{k\tau|k\tau}$ are simulated in a way to be non-random, $Q_{\hat{\mathbf{c}}_{k\tau|k\tau}} = 0$. In the next section, the performance of the proposed user-filter is studied for a real-world GNSS data-set for which the equality $Q_{\hat{\mathbf{c}}_{k\tau|k\tau}} = 0$ does not hold. Instead, the filter has to rely on the approximate counterpart $Q_{\hat{\mathbf{c}}_{k\tau|k\tau}} \approx 0$.

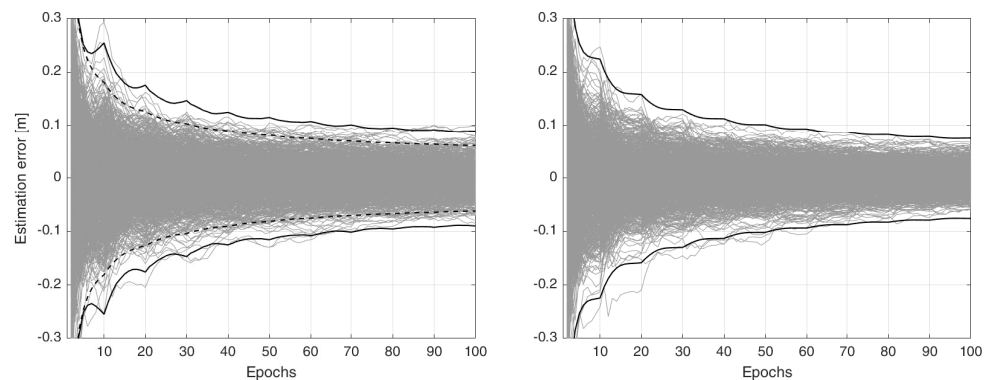


Figure 4. Time-series of the ionospheric estimation-error $b_h - \hat{b}_{h|h}$ ($h = 1, \dots, 100$) of Example 2 (grey lines). Given 1000 normally distributed samples, the misspecified user-filter computes filtered ionospheric solutions under two cases. **(Left):** Case 2 (13), in which only the variance of the corrections is considered. **(Right):** Case 3 (Figure 3), in which both the variance and time-correlation of the corrections are incorporated by the proposed filter. The incorrect 99.9% confidence intervals (dashed lines), reported by the user-filter, are compared with their correct versions (black solid lines).

5. A Single-Station PPP-RTK Example

In this section, we make use of a Galileo dual-frequency (E1/E5a) data-set to study the positioning performance of the misspecified user-filter for three variants: Case 1 (10), Case 2 (13), and Case 3 (Figure 3). The data-set was collected with a 1 Hz sampling-rate on 21 January 2022 by two permanent GNSS stations: CUT0 and UWA0, both located in Western Australia. The precise orbital corrections are a priori applied to the data.

State-space corrections (i.e., clock, ionospheric and phase-bias corrections) are generated via a single-station PPP-RTK setup [7,13]. The setup is visualized in Figure 5. Station CUT0 serves as correction-provider, whereas the station UWA0 serves as a user that is approximately 8 km away from the provider. To emphasize the performance of our filter formulation (Case 3) in handling time-delayed corrections, we consider rather high correction latencies. Accordingly, the clock correction packs are assumed to be made available to the user every 10 s, ionospheric correction packs every 30 s, and phase-bias correction packs every 10 min. In order to make the approximation $Q_{\hat{c}_{k\tau|k\tau}} \approx 0$ plausible, the provider sends the corrections only after 1 h from the time of their filter-initialization. The user then time-predicts each correction and runs their filter over 20 min (1200 epochs). To form their observation variance matrix, the user employs the sinusoidal satellite elevation-weighting strategy with the zenith-referenced standard-deviations of 20 cm and 2 mm for their undifferenced code and phase measurements, respectively.

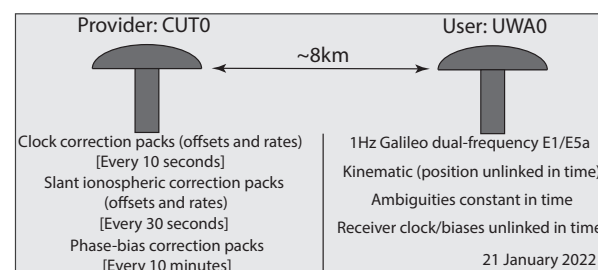


Figure 5. Single-station PPP-RTK setup for the experimental data-set. The temporal behaviour of the satellite clocks and ionospheric delays is modelled by a constant-velocity dynamic model with process noises' standard deviations of $3 \text{ mm}/\sqrt{s^3}$ and $1 \text{ mm}/\sqrt{s^3}$, respectively. The satellite phase biases are assumed to follow a random-walk (constant-state) process with a standard deviation of $0.01 \text{ cyc.}/\sqrt{s}$.

Over the fixed 1200 epochs, we consider both the ambiguity-float and -fixed positioning performance of the user. To show the role played by the correction latency, we also compare the user positioning results with those obtained with *zero* latency. Let us first focus on the ambiguity-float results, i.e., before resolving the user float ambiguities to their integers. The results are presented in Figure 6. The left-panel corresponds to the case where no correction latency is experienced by the user. For the zero-latency case, the results of Cases 1 and 2 *coincide* with those of Case 3 (green lines). This can be understood as follows. Recall that Case 1 (red lines) treats the corrections as non-random, and Case 2 (blue lines) models the variance of the corrections, while Case 3 models both the variance and time-correlation of the corrections. As both Cases 2 and 3 rely on the approximation $Q_{\hat{c}_{k\tau}|k\tau} \approx 0$, they deliver results identical to those of Case 1 in the case of zero latency (i.e., when $i = k\tau$). Thus, the underlying difference of these three cases only lies in the way of correctional uncertainty due to the fact that the latency is handled.

The effect of nonzero correction latency is shown in the right-panel of Figure 6. Since $i \geq k\tau$, the user has to time-predict the corrections, thereby requiring that it take the associated correctional uncertainty into account. As Case 1 does not consider such uncertainty, it delivers positioning results with the largest root-mean-squared (RMS) errors in all the three East–North–Up directions. For the horizontal components, Case 2 delivers similar results to those of Case 3 in an RMS sense. However, Case 3 experiences less discontinuities in its positioning results than Case 2. Such discontinuities are owed to the *periodic* increase in the latency that it varies from 0 to $\tau - 1$ seconds for every correction pack.

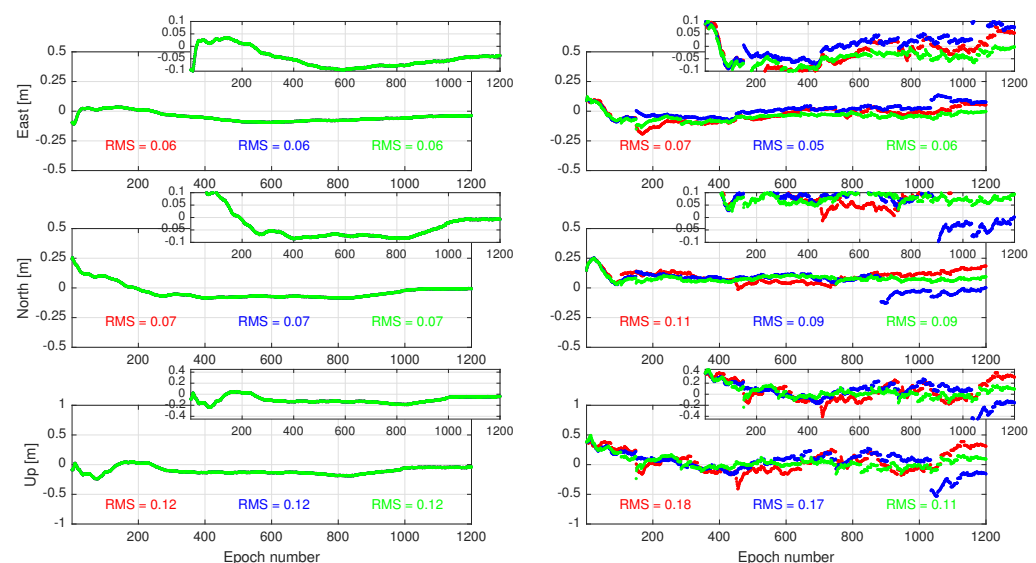


Figure 6. Time-series of the **ambiguity-float** positioning errors in the East–North–Up directions when the user experiences no correction latency (**left**) and when the correction packs are provided with a time-delay (**right**). The results of Cases 1, 2, and 3 are indicated in red, blue, and green, respectively.

Let us now turn our attention to the user ambiguity-fixed results, i.e., after mapping the float ambiguities to their integers. At every epoch, full ambiguity resolution using the integer least-squares estimation is conducted *without* an extra ambiguity validation procedure [25]. The results are presented in Figure 7. As before, both the zero-latency (left-panel) and nonzero-latency (right-panel) cases are considered. While the three cases 1, 2, and 3 perform the same when the user receives corrections with no time-delay, their performances become distinct when treating time-delayed corrections. Case 1 (red lines) exhibits several large ambiguity-resolved positioning errors (≥ 1 dm) in all three East–North–Up directions, leading to large positioning RMS errors. In other words, the corresponding ambiguity success-rate is not sufficiently high to deliver successful ambiguity-resolved positioning solutions. On the contrary, Cases 2 and 3 outperform Case 1 in the RMS sense.

Similar to their ambiguity-float counterparts, the RMS errors of Cases 2 and 3 are the same for the horizontal components.

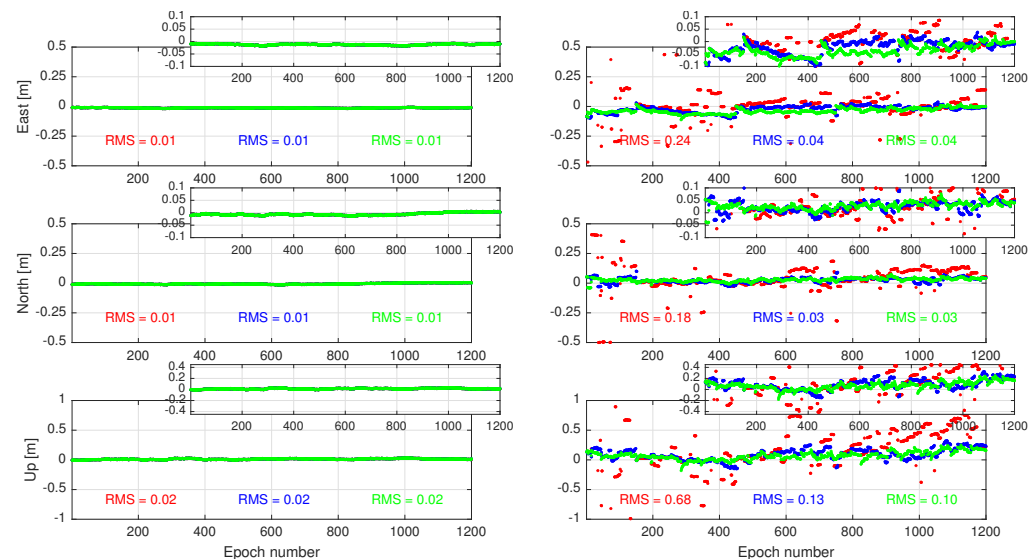


Figure 7. Time-series of the **ambiguity-fixed** positioning errors in East–North–Up directions when the user experiences no correction latency (**left**) and when the correction packs are provided with a time-delay (**right**). The results of Cases 1, 2, and 3 are indicated in red, blue, and green, respectively.

Observing the results in Figures 6 and 7, one may be inclined to conclude that the performance of the proposed user-filter (Case 3) is the same as that of Case 2. From a computational point of view, Case 2 seems to be more attractive as it does not need the augmentation of the user state-vector with the corrections. This is in contrast to the formulation of Case 3 which needs to recursively update both the user and correction parameter solutions. One should, however, remark that those results only represent one *realization* of the user-filter’s parameter solutions. In order to infer the overall performance of the user-filter under the formulations offered by Cases 1, 2, and 3, one needs to instead consider the *distribution* of the user parameter solutions. To that end, we generate 300 different realizations of the results by shifting the user-filter starting epoch i every 15 s. The time-series of the medians (i.e., 50% percentiles) of these realizations are presented in Figure 8 for both the user ambiguity-float (left) and -fixed (right) cases. The medians of the positioning errors corresponding to Cases 2 and 3 are shown to be considerably smaller than those of Case 1. The results also indicate that Case 3 outperforms Case 2 as it, on average, delivers smaller medians of the positioning errors. Note also the presence of periodic jumps of the medians for all the three cases. This behaviour is due to the periodic nature of the correction latencies that vary from zero to $\tau - 1$ s for each correction pack. It is, however, observed that Case 3 exhibits smaller periodic jumps. This is because Case 3 accounts for the uncertainty of the time-delayed corrections by modelling their variance and time-correlation through their dynamic model (5).

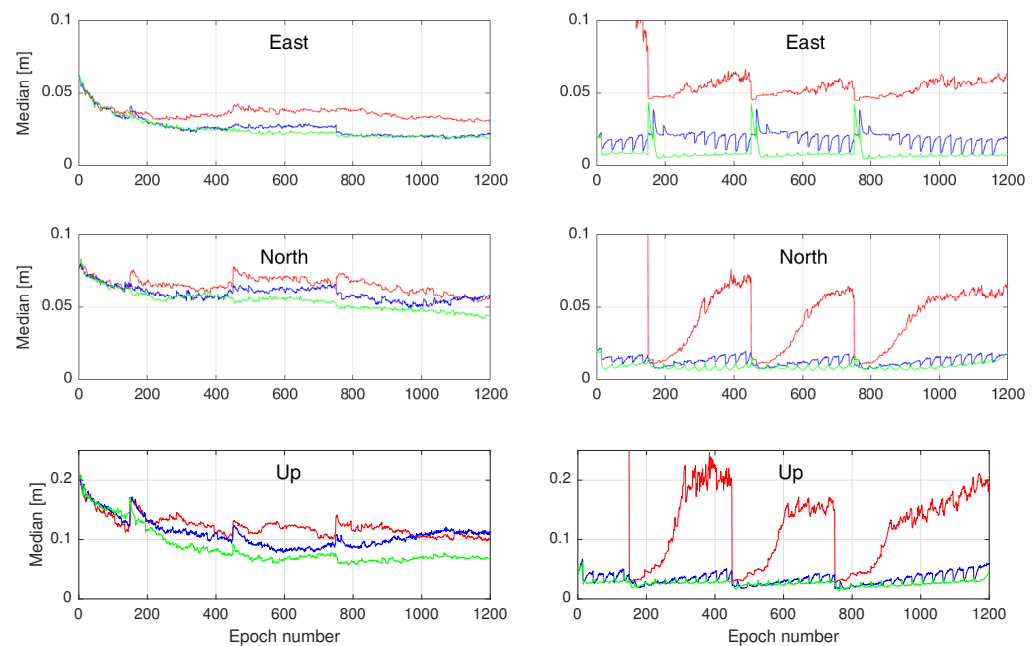


Figure 8. Medians (50% percentiles) of the absolute positioning errors corresponding to 300 user-filter realizations before (left) and after (right) ambiguity-fixing in the East–North–Up directions when the correction packs are provided with a time-delay (cf. Figure 5). The results of Cases 1, 2, and 3 are indicated in red, blue, and green, respectively.

6. Conclusions and Outlook

In this contribution, a new multi-epoch formulation for the PPP-RTK user-filter was proposed (Figure 3). Under the proposed formulation, the user-filter state-vector b_h is augmented with the correction parameter vector c_h as $[b_h^T, c_h^T]^T$, thereby allowing the corrections to be *jointly* measurement-updated with the user parameter solutions.

It was shown why the user-filter always remains misspecified, and therefore, *suboptimal* in the minimum-variance sense, no matter which formulation is adopted. Since the PPP-RTK provider has the freedom to disseminate their filtered corrections $\hat{c}_{k\tau|k\tau}$ only after a sufficiently long period from the time of their filter-initialization, the user can, however, benefit from the approximation $Q_{\hat{c}_{k\tau|k\tau}} \approx 0$ to limit the sub-optimality level of their filter. Instead of current formulations in which the time-correlated corrected observation vectors $\underline{u}_h - C_h \hat{c}_{h|k\tau}$ ($h = i, i + 1, \dots$) serve as input of the user-filter, our proposed formulation treats the corrections $\hat{c}_{h|k\tau}$ as additional observation vectors and works with the augmented observation vector $[\underline{u}_h^T, \hat{c}_{h|k\tau}^T]^T$. Relying on the approximation $Q_{\hat{c}_{k\tau|k\tau}} \approx 0$, the observation vectors $[\underline{u}_h^T, \hat{c}_{h|k\tau}^T]^T$ ($k = 1, 2, \dots$) would then become time-uncorrelated, leading the user-filter to deliver close-to-minimum-variance solutions. Under the proposed formulation, however, the user needs to know the *dynamic model* that the provider uses for the corrections. For further standardization of the State Space Representation (SSR) corrections (<https://www.igs.org/formats-and-standards> (accessed on 18 December 2022)), Radio Technical Commission for Maritime Services (RTCM) committees may therefore request PPP-RTK providers to disclose the dynamic models underlying their SSR corrections.

A single-station PPP-RTK setup was employed to numerically demonstrate the superiority of the proposed filter over existing formulations. To emphasize the performance of our formulation, rather high correction latencies were considered, e.g., ionospheric correction packs were provided every 30 s. It was observed that the proposed filter delivers smaller estimation errors (in an RMS sense) when handling time-delayed corrections.

In the present contribution, attention was focused on the formulation of the user-filter only. Addressing open research questions such as ‘how to develop measures for assessing

the approximation $Q_{\hat{c}_{k\tau|k\tau}} \approx 0'$, and ‘the extent to which such approximation drives the sub-optimality level of the user-filter’ are topics of future works.

Author Contributions: Conceptualization, A.K., P.J.G.T. and D.P.; Formal analysis, A.K.; Writing—original draft, A.K.; Writing—review & editing, P.J.G.T. and D.P. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: The GNSS data used in this contribution are freely accessible via <http://saegnss2.curtin.edu.au/ldc/> (accessed on 18 December 2022) and <https://cddis.nasa.gov/archive/gnss/data/> (accessed on 18 December 2022).

Acknowledgments: This work was conducted as part of the first author’s Alexander von Humboldt Fellowship at the Institute for Communication and Navigation, the German Aerospace Center (DLR), Munich, with Christoph Günther and Gabriele Giorgi as their hosts. This support is gratefully acknowledged.

Conflicts of Interest: The authors declare no conflict of interest.

Appendix A. Supplementary Proofs

Proof of (7). Combining the TU error $c_{i+1} - \hat{c}_{i+1|i} = \Phi^c(c_i - \hat{c}_{i|i}) + \underline{w}_{i+1}^c$ with its MU counterpart $c_{i+1} - \hat{c}_{i+1|i+1} = (I - K_{i+1}^c A_{i+1})(c_{i+1} - \hat{c}_{i+1|i}) + K_{i+1}^c \underline{e}_{i+1}$ establishes a link between the two successive MU errors $\hat{c}_i = c_i - \hat{c}_{i|i}$ and $\hat{c}_{i+1} = c_{i+1} - \hat{c}_{i+1|i+1}$ as follows

$$\hat{c}_{i+1} = [(I - K_{i+1}^c A_{i+1})\Phi^c] \hat{c}_i + \underbrace{(I - K_{i+1}^c A_{i+1}) \underline{w}_{i+1}^c + K_{i+1}^c \underline{e}_{i+1}}_{L_{i,i+1}} \quad (\text{A1})$$

where the random quantity $L_{i,i+1}$ is a function of the process and measurement noises \underline{w}_{i+1}^c and \underline{e}_{i+1} . The repeated application of (A1) for the remaining successive epochs $i + q + 1$ and $i + q + 2$ ($q = 1, \dots, j - i - q - 2$) gives

$$\hat{c}_j = \left[\prod_{h=i+1}^j (I - K_h^c A_h) \Phi^c \right] \hat{c}_i + L_{i,i+1,\dots,j}, \quad j > i \quad (\text{A2})$$

where the random quantity $L_{i,i+1,\dots,j}$ is a function of the process and measurement noises \underline{w}_{i+q}^c and \underline{e}_{i+q} ($q = 1, \dots, j - i$). Since the MU error \hat{c}_i is *uncorrelated* with the process and measurement noises of the upcoming epochs $h = i + 1, \dots, j$ [26], it is also uncorrelated with their linear function $L_{i,i+1,\dots,j}$. Therefore, an application of the covariance propagation law to (A2), together with the definition $Q_{\hat{c}_{i|i}} := \text{Cov}(\hat{c}_i, \hat{c}_i)$, gives (7). \square

Proof of (8). From the TU errors

$$c_{k\tau+q} - \hat{c}_{k\tau+q|k\tau+q-1} = \Phi^c(c_{k\tau+q-1} - \hat{c}_{k\tau+q-1|k\tau+q-1}) + \underline{w}_{k\tau+q}^c, \quad q = 1, \dots, i - k\tau, \quad (\text{A3})$$

follows that

$$c_i - \hat{c}_{i|k\tau} = \Phi_{(i-k\tau)}^c(c_{k\tau} - \hat{c}_{k\tau|k\tau}) + \sum_{h=k\tau+1}^i \Phi_{(i-h)} \underline{w}_h^c. \quad (\text{A4})$$

The application of the covariance propagation law to (A4), together with $\text{Cov}(c_{k\tau} - \hat{c}_{k\tau|k\tau}, \underline{w}_h^c) = 0$ ($h = k\tau + 1, \dots, i$), gives (8). \square

References

1. Wubben, G.; Schmitz, M.; Bagge, A. PPP-RTK: Precise Point Positioning using state-space representation in RTK networks. In Proceedings of the ION GNSS, Long Beach, CA, USA, 13–16 September 2005; pp. 13–16.
2. Laurichesse, D.; Mercier, F. Integer ambiguity resolution on undifferenced GPS phase measurements and its application to PPP. In Proceedings of the 20th International Technical Meeting of the Satellite Division of The Institute of Navigation (ION GNSS 2007), Fort Worth, TX, USA, 25–28 September 2007; pp. 839–848.

3. Collins, P.; Bisnath, S.; Lahaye, F.; Heroux, P. Undifferenced GPS ambiguity resolution using the decoupled clock model and ambiguity datum fixing. *Navigation* **2010**, *57*, 123–135. [\[CrossRef\]](#)
4. Teunissen, P.J.G.; Odijk, D.; Zhang, B. PPP-RTK: Results of CORS Network-Based PPP with Integer Ambiguity Resolution. *J. Aeronaut. Astronaut. Aviat.* **2010**, *42*, 223–229.
5. Geng, J.; Bock, Y. Triple-frequency GPS precise point positioning with rapid ambiguity resolution. *J. Geod.* **2013**, *87*, 449–460. [\[CrossRef\]](#)
6. Wang, K.; Khodabandeh, A.; Teunissen, P.J.G. A study on predicting network corrections in PPP-RTK processing. *Adv. Space Res.* **2017**, *60*, 1463–1477. [\[CrossRef\]](#)
7. Khodabandeh, A. Single-station PPP-RTK: Correction latency and ambiguity resolution performance. *J. Geod.* **2021**, *95*, 1–24. [\[CrossRef\]](#)
8. Odijk, D.; Teunissen, P.J.G.; Khodabandeh, A. Single-Frequency PPP-RTK: Theory and Experimental Results. *IAG Symp.* **2014**, *139*, 167–173.
9. Psychas, D.; Khodabandeh, A.; Teunissen, P.J.G. Impact and mitigation of neglecting PPP-RTK correctional uncertainty. *GPS Solut.* **2022**, *26*, 33. [\[CrossRef\]](#)
10. Teunissen, P.J.G.; Khodabandeh, A.; Psychas, D. A generalized Kalman filter with its precision in recursive form when the stochastic model is misspecified. *J. Geod.* **2021**, *95*, 108. [\[CrossRef\]](#)
11. Kalman, R.E. A new approach to linear filtering and prediction problems. *J. Basic Eng.* **1960**, *82*, 35–45. [\[CrossRef\]](#)
12. Simon, D. *Optimal State Estimation: Kalman, H [Infinity] and Nonlinear Approaches*; John Wiley and Sons: Hoboken, NJ, USA, 2006.
13. Khodabandeh, A.; Teunissen, P.J.G. An analytical study of PPP-RTK corrections: Precision, correlation and user-impact. *J. Geod.* **2015**, *89*, 1109–1132. [\[CrossRef\]](#)
14. Leick, A.; Rapoport, L.; Tatarnikov, D. *GPS Satellite Surveying*, 4th ed.; John Wiley and Sons: Hoboken, NJ, USA, 2015.
15. Odijk, D.; Zhang, B.; Khodabandeh, A.; Odolinski, R.; Teunissen, P.J.G. On the estimability of parameters in undifferenced, uncombined GNSS network and PPP-RTK user models by means of S-system theory. *J. Geod.* **2015**, *90*, 15–44. [\[CrossRef\]](#)
16. Teunissen, P.J.G.; Montenbruck, O. (Eds.) *Springer Handbook of Global Navigation Satellite Systems*; Springer: Berlin/Heidelberg, Germany, 2017.
17. Zhang, B.; Liu, T.; Yuan, Y. GPS receiver phase biases estimable in PPP-RTK networks: Dynamic characterization and impact analysis. *J. Geod.* **2018**, *92*, 659–674. [\[CrossRef\]](#)
18. Teunissen, P.J.G. *Dynamic Data Processing*; Series on Mathematical Geodesy and Positioning; Delft University Press: Delft, The Netherlands, 2001.
19. Teunissen, P.J.G. *Adjustment Theory: An Introduction*; Series on Mathematical Geodesy and Positioning; Delft University Press: Delft, The Netherlands, 2000.
20. Vollath, U.; Talbot, N.C.; Glocker, M.; Chen, X.; Leandro, R. GNSS Signal Processing with Delta Phase. WO2012151006 A1, 8 November 2012.
21. Laurichesse, D. The CNES Real-time PPP with undifferenced integer ambiguity resolution demonstrator. In Proceedings of the 24th International Technical Meeting of The Satellite Division of the Institute of Navigation (ION GNSS 2011), Portland, OR, USA, 20–23 September 2011; pp. 654–662.
22. Leandro, R.F.; Santos, M.C.; Langley, R.B. Analyzing GNSS data in precise point positioning software. *GPS Solut.* **2011**, *15*, 1–13. [\[CrossRef\]](#)
23. Maybeck, P. *Stochastic Models, Estimation, and Control*; Academic Press: Cambridge, MA, USA, 1979; Volume 1; Republished in 1994.
24. Petovello, M.G.; O’Keefe, K.; Lachapelle, G.; Cannon, M.E. Consideration of time-correlated errors in a Kalman filter applicable to GNSS. *J. Geod.* **2009**, *83*, 51–56. [\[CrossRef\]](#)
25. Verhagen, S.; Teunissen, P.J.G. The ratio test for future GNSS ambiguity resolution. *GPS Solut.* **2013**, *17*, 535–548. [\[CrossRef\]](#)
26. Teunissen, P.J.G.; Khodabandeh, A. BLUE, BLUP and the Kalman filter: Some new results. *J. Geod.* **2013**, *87*, 461–473. [\[CrossRef\]](#)

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.