Improving the Speed of CORS Network RTK Ambiguity Resolution

Dennis Odijk
GNSS Research Centre
Curtin University of Technology
Perth, Australia
d.odijk@curtin.edu.au

Peter J.G. Teunissen
GNSS Research Centre
Curtin University of Technology
Perth, Australia
p.teunissen@curtin.edu.au
Delft Institute of Earth Observation and Space Systems
Delft University of Technology
Delft, The Netherlands
p.j.g.teunissen@tudelft.nl

Abstract—**Carrier-phase ambiguity resolution between CORS Network RTK reference stations is hampered because of the presence of differential ionospheric delays. As a consequence, after a (power) failure or when a new satellite rises above the cut-off elevation, the float ambiguity solution needs to converge before the integer ambiguities can be reliably estimated. These convergence times can be up to tens of minutes, depending on the distance between the CORS stations and the ionospheric conditions. In this paper we propose a processing strategy to improve the speed of CORS Network RTK ambiguity resolution based on current dual-frequency GPS. This strategy suggests to use the ionosphere-weighted model, incorporating the known CORS station positions and a priori information on the ionospheric delays, in combination with the LAMBDA method and Fixed Failure-rate Ratio Test.**

Keywords-GNSS, CORS Network RTK, Ambiguity Resolution, LAMBDA method, Fixed Failure-rate Ratio Test, convergence time

I. Introduction

This contribution focuses on the carrier-phase ambiguity resolution performance of Continuously Operating GNSS Reference Stations (CORS). The speed with which successful ambiguity resolution is possible at the CORS network sites, is essential for the whole processing chain that culminates in the generated ionospheric corrections at the user (rover) site. These ionospheric corrections are meant to make Real-Time Kinematic (RTK) positioning at the rover site possible by eliminating or significantly reducing the differential ionospheric delays between rover and network (Network RTK), see also Figure 1.

The rover ionospheric corrections are obtained from a spatiotemporal prediction that uses the network ionospheric delays as input. To be able to generate precise ionospheric corrections in a timely fashion, it is a prerequisite that the network ionospheric delays are based on fast and successfully resolved network integer carrier phase ambiguities. Unfortunately, at present, instantaneous network ambiguity resolution is not yet feasible, even though the a priori known positions of the network stations are used. This nontrivial issue

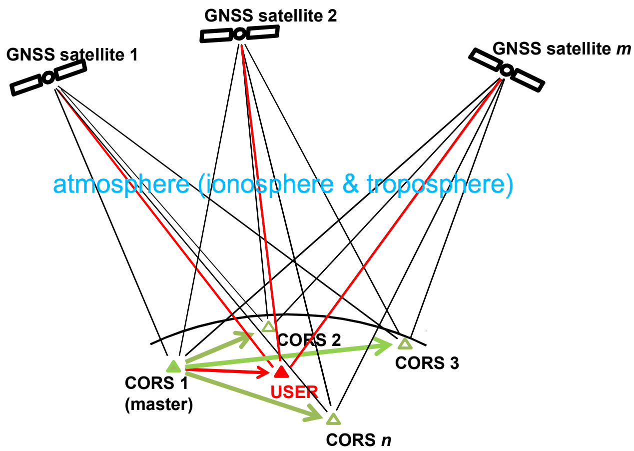

Figure 1 Schematic principle of CORS Network RTK: data of a CORS network station and corrections determined from the network are disseminated to an RTK user.

of real-time network ambiguity resolution has, among others, already been emphasized in [1].

In order to resolve the network integer ambiguities reliably, the float solution should have sufficiently converged and this convergence time can take up to tens of minutes, depending on the baseline length and the atmospheric conditions. Also when new satellites appear over the horizon, some time is required before the integer ambiguities of these rising satellites can be resolved. Similarly, CORS station power failures will also necessitate (partial) re-initializations of the Kalman filter, thus causing convergence times as well.

In [2] by means of an evaluation of the Ambiguity Dilution Of Precision (ADOP, see [3]) the bottlenecks of fast CORS Network RTK ambiguity resolution have been identified. These comprise a lack of number of satellites and number of frequencies and the absence of ionospheric information. In the present contribution we will focus on how CORS Network RTK ambiguity resolution can be speeded up in the current dual-frequency GPS case by means of adding a priori information of the double-differenced (DD) ionospheric delays. The emphasis will be on the ambiguity resolution process itself, which should consist of the LAMBDA method in combination with the Fixed Failure-rate Ratio Test. Numerical examples will be given for GPS data of CORS networks with inter station distances of up to 120 km. In this context it is remarked that in this paper we will only focus on ambiguity resolution between the CORS network stations; CORS Network RTK user ambiguity resolution is *not* taken into account.

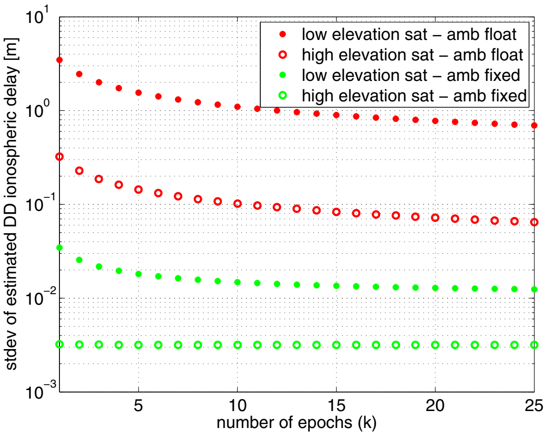

Figure 2 Example of standard deviation of CORS Network DD ionospheric delays before and after ambiguity fixing.

As example to stress the importance of CORS Network RTK ambiguity resolution, Figure 2 depicts the standard deviation of two DD ionospheric delays as function of the number of epochs, based on float as well as fixed ambiguities. The first DD ionospheric delay concerns a satellite with relatively low elevation, while the second DD ionospheric delay corresponds to a relatively high elevation satellite. For both DD ionospheric delays an identical reference satellite is used, having a relatively high elevation. As can be seen from Figure 2, the precision of the float DD ionospheric delays is at dm-level for the double-difference with high elevation, and even worse at m-level for the double-difference with low elevation. A dramatic improvement can however be observed if the integer ambiguities are fixed. The precision of the DD ionospheric delay with low elevation improves to cm-level, while that of the DD ionospheric delay with high elevation reaches the level of only a few mm.

II. CORS Network RTK Processing Model

A. *Geometry-free Linear Combination or Ionosphere-float Model for CORS GNSS Data Processing*

The DD observations equations for GPS phase and code data are well known and can be given as [4]:

$$\begin{aligned} \phi_1 &= \rho + \tau - \mu_1 \iota_1 + \lambda_1 a_1 + \varepsilon_1 \\ \phi_2 &= \rho + \tau - \mu_2 \iota_1 + \lambda_2 a_2 + \varepsilon_2 \\ p_1 &= \rho + \tau + \mu_1 \iota_1 + e_1 \\ p_2 &= \rho + \tau + \mu_2 \iota_1 + e_2 \end{aligned} \qquad (1)$$

where ϕ_1 and ϕ_2 denote the vector of DD phase observables on L1 and L2 respectively, p_1 and p_2 their DD code counterparts, ρ the DD receiver-satellite ranges, τ the DD tropospheric delays, ι_1 the DD ionospheric delays on L1, $\mu_1 = 1$ and $\mu_2 = \lambda_2^2 / \lambda_1^2$ the ionospheric coefficients on L1 and L2, a_1 and a_2 the DD ambiguities on L1 and L2 respectively, λ_1 and λ_2 the L1 and L2 wavelengths, ε_1 and ε_2 the random errors for the phase observations on L1 and L2, respectively, and e_1 and e_2 the random errors for the code observations on L1 and L2, respectively.

Usually for the purpose of CORS ionospheric delay estimation, as described in the literature, the well-known geometry-free linear combination of L1 and L2 is taken, e.g. [5]-[6]:

$$\begin{aligned} \phi_1 - \phi_2 &= -(\mu_1 - \mu_2)\iota_1 + \lambda_1 a_1 - \lambda_2 a_2 + \varepsilon_1 - \varepsilon_2 \\ p_1 - p_2 &= (\mu_1 - \mu_2)\iota_1 + e_1 - e_2 \end{aligned} \qquad (2)$$

As can be seen from (2), both the receiver-satellite ranges as well as the tropospheric delays have been eliminated. However, this also implies that no use is made of the *known* positions of the CORS stations when the ambiguities are resolved on basis of the geometry-free combinations. The use of this information however will speed up CORS network ambiguity resolution.

An approach that does take the known CORS station positions into account is the following. Since the DD receiver-satellite ranges become known based on the known satellite and station positions, they can be subtracted from the DD phase and code observations. In addition, the DD tropospheric delays are mapped to zenith tropospheric delays, one per station relative to the CORS master station. Then the following model is solved, written using a Gauss-Markov formulation:

$$E\left(\begin{pmatrix} \phi_1 - \rho \\ \phi_2 - \rho \\ p_1 - \rho \\ p_2 - \rho \end{pmatrix}\right) = \begin{pmatrix} M & -\mu_1 I & \lambda_1 I & \\ M & -\mu_2 I & & \lambda_2 I \\ M & \mu_1 I & & \\ M & \mu_2 I & & \end{pmatrix} \begin{pmatrix} \tau_z \\ \iota_1 \\ a_1 \\ a_2 \end{pmatrix} \qquad (3)$$

with $E(\cdot)$ the expectation operator, I the identity matrix, τ_z the relative zenith tropospheric delays and M their mapping function coefficients written in matrix form. The model (3) is known as the (station-fixed) *ionosphere-float* model [7]. The corresponding stochastic model of the observations can be given as follows:

$$D\left(\begin{pmatrix} \phi_1 - \rho \\ \phi_2 - \rho \\ p_1 - \rho \\ p_2 - \rho \end{pmatrix}\right) = \begin{pmatrix} c_{\phi_1}^2 & & & \\ & c_{\phi_2}^2 & & \\ & & c_{p_1}^2 & \\ & & & c_{p_2}^2 \end{pmatrix} \otimes (D_n^T D_n) \otimes (D_m^T W_m^{-1} D_m) \qquad (4)$$

with $D(\cdot)$ the dispersion operator, $c_{\phi_1}^2$ and $c_{\phi_2}^2$ the variance factors of the undifferenced phase observations on L1 and L2, respectively, $c_{p_1}^2$ and $c_{p_2}^2$ the variance factors of the undifferenced code observations on L1 and L2, respectively, D_n the between-receiver difference operator, D_m the between-satellite difference operator, W_m a weight matrix to model satellite elevation dependency and $\otimes$ the matrix Kronecker product.

The ionosphere-float model is usually solved in three steps: float solution – ambiguity resolution – fixed solution. These steps will be explained in the next three subsections. Figure 3 schematically depicts the three-step procedure.

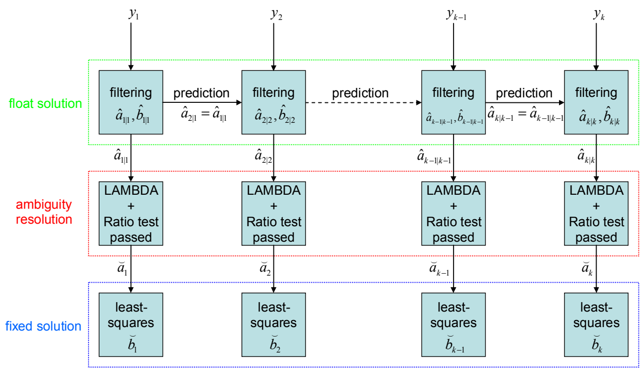

Figure 3 Three-step procedure for the processing of CORS Network data.

B. *Float Solution: Kalman Filter Implementation*

To compute the float solution, the integer constraints to the DD ambiguities are discarded. In order to obtain this solution in real-time, for CORS Network data processing the float solution is recursively computed by means of a Kalman filter implementation. The state vector of the Kalman filter then consists of the parameters as given in (3), where we make a distinction between the DD ambiguities, denoted in a vector $a=(a_1^T, a_2^T)^T$ and all other parameters, i.e. the zenith tropospheric delays and the DD ionospheric delays, collected in vector $b=(\tau_z^T, \iota_1^T)^T$. In the time update of the Kalman filter use is made of the time-constant property of the phase ambiguities (provided that no cycle slips occur).

C. *Ambiguity Resolution: LAMBDA Method and Fixed Failure-Rate Ratio Test*

The second step towards precise CORS-based ionospheric delay estimation consists of ambiguity resolution. The float ambiguity solution, which is obtained in real time from the Kalman filter, is input to the LAMBDA method [8]. If we collect the dual-frequency DD ambiguities into one vector, and denote its float solution as $\hat{a}$, with variance-covariance matrix $Q_{\hat{a}}$, the integer least-squares mapping as implemented in LAMBDA can be written as:

$$\breve{a} = S(\hat{a}); \quad S: \; R^n \rightarrow Z^n \tag{5}$$

where $\breve{a}=(\breve{a}_1^T, \breve{a}_2^T)^T$ denotes the integer least-squares solution. It is emphasized that no linear combinations of ambiguities are formed a priori in order to enhance the ambiguity search (such as the wide lane, etc.). The decorrelating transformation inside the LAMBDA method determines the optimal ambiguity combinations.

Since the LAMBDA method always outputs an integer solution, we have to decide whether this integer solution can be accepted as the correct integer solution with a sufficient amount of reliability. For this purpose the Ratio Test is carried out, which tests whether the quadratic norm based on the integer least-squares solution, denoted as $\|\hat{a}-\breve{a}\|^2_{Q_{\hat{a}}}$, with $\|\cdot\|^2_Q = (\cdot)^T Q^{-1}(\cdot)$, is sufficiently smaller than the quadratic norm based in the second-best integer least-squares solution, which is denoted as $\|\hat{a}-\breve{a}_{(2)}\|^2_{Q_{\hat{a}}}$. The Ratio Test is thus defined as:

$$\text{Reject } \breve{a} \text{ if } \frac{\|\hat{a}-\breve{a}\|^2_{Q_{\hat{a}}}}{\|\hat{a}-\breve{a}_{(2)}\|^2_{Q_{\hat{a}}}} > \mu, \text{ with } 0 < \mu \leq 1 \tag{6}$$

where μ denotes the threshold or critical value of the Ratio Test. Traditionally, this threshold is taken as a fixed value, e.g. 1/2 or 1/3. The Fixed Failure-rate (FF) Ratio Test differs from the traditional ratio test in the sense that no fixed critical value is used [9]. However, the critical value is set based on the GNSS model at hand such that the probability of accepting a wrong integer solution (i.e. the failure rate) is below a fixed user-defined threshold (e.g. 0.001).

D. *Fixed Solution: Single-Epoch Least-Squares*

Once the integer solution is accepted, the third processing step is executed, in which the solution for the atmospheric parameters is improved, since the integer ambiguities are fixed:

$$\breve{b} = \hat{b} - Q_{\hat{b}\hat{a}} Q_{\hat{a}}^{-1}(\hat{a}-\breve{a}) \tag{7}$$

with $Q_{\hat{b}\hat{a}}$ the covariance matrix between the float tropospheric/ionospheric delays and the ambiguities. The fixed solution for tropospheric and ionospheric delay, denoted as $\breve{b}=(\breve{\tau}_z^T, \breve{\iota}_1^T)^T$ is now driven by the high quality of the phase data.

III. CORS NETWORK RTK AMBIGUITY RESOLUTION: CONVERGENCE TIMES

In our implementation of the CORS Network RTK data processing procedure, see Figure 3, the ambiguity resolution (AR) consisting of LAMBDA and FFRatio Test is repeated for every epoch. Generally, a number of epochs or time span is required before the FFRatio Test is passed after beginning of operation or after a new satellite has risen above the cut-off elevation, since the float ambiguity solution has to converge. In our approach also after the ambiguities have converged and the FFRatio Test is passed the CORS Network AR is repeated for every epoch, and this is to gain confidence that indeed the correct integer solution is estimated as time increases by comparing the integer solution of the current epoch with the previous epoch. A similar approach is described in [10].

Thus, it is clear that CORS Network AR convergence times are driven by the outcomes of the FFRatio Test. As an illustration, let us first investigate the consequences when this FFRatio test is *not* used. Thus, the LAMBDA method estimates integer solution for all epochs, but it is not verified whether these can be accepted or not. In Figure 4 it is shown that for some epochs the wrong integers are estimated for some satellites as soon as they are included in the processing (after they have risen). After a number of epochs of wrong integer estimation, the correct integer solution is however obtained.

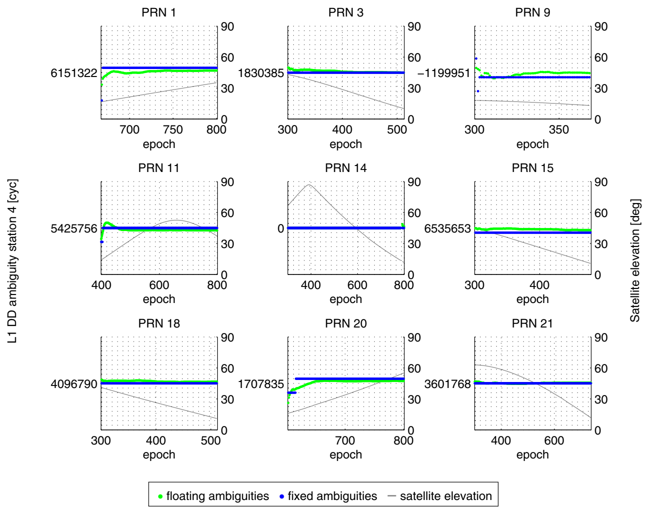

Figure 4 Consequences of not performing the FFRatio Test: wrong integer estimation for satellites PRNs 9, 11 and 20 at the beginning of their appearance. The green curve is the float ambiguity solution from the Kalman filter, while the blue curve represents the integer solution estimated each epoch by LAMBDA.

If the FFRatio Test is turned on, the consequence is that after the inclusion of a newly risen satellite the FFRatio Test is not accepted. Although the execution of the FFRatio Test protects against wrong integer estimation, the full integer vector including existing ambiguities that were already accepted before, is not accepted as soon as a new satellite is included. See Figure 5, which shows the number of satellites, ratios and the outcomes of the FFRatio Tests during a time span of 18.3 hours (data sampling: 30 sec) for a CORS network consisting of 5 stations of the Southern California Integrated GPS Network (SCIGN), having inter-station distances of 70-120 km. Data of 11 April 2001 has been used, a date close to the most recent solar max, causing the GPS data to be biased by a considerable amount of ionospheric delays. The failure rate has been fixed to 0.001 (so for only 1 out of 1000 epochs we accept that the FFRatio Test accepts a wrong integer solution). As can be seen from the figure, the FFRatio Test is not passed for a long time (up to 1 hour) after a newly risen satellite is included in the processing. This is of course not acceptable and the solution to this is *partial* fixing, i.e. the data of the new satellite are allowed to contribute to the float solution, but its integer ambiguities are not fixed; only those of the 'old' ambiguities. Parallel to this partial ambiguity fixing, the LAMBDA method is employed to the full ambiguity vector (i.e. 'old' and 'new') but purely for the purpose to decide whether the ambiguities of the new satellite have converged sufficiently such that they can be integrated together with the old satellite ambiguities in a full fixing with LAMBDA. This decision should thus be based on the outcome of the FFRatio Test; if not passed, the next epoch this should be repeated up to the moment the full integer vector is accepted. In Figure 6, instead of taking the outcome of the FFRatio Test as the decision to include the new satellite into the ambiguity fixing of the complete ambiguity vector, a fixed convergence time of 20 minutes has been used after a new satellite has risen and before the data of this new satellite in the ambiguity resolution of the complete network. As can be seen from the figure, this 20-minute convergence time is representative for most of the time span of 18.3 hours; only for 3% of the time the FFRatio Test is still not passed and the actual convergence time is longer.

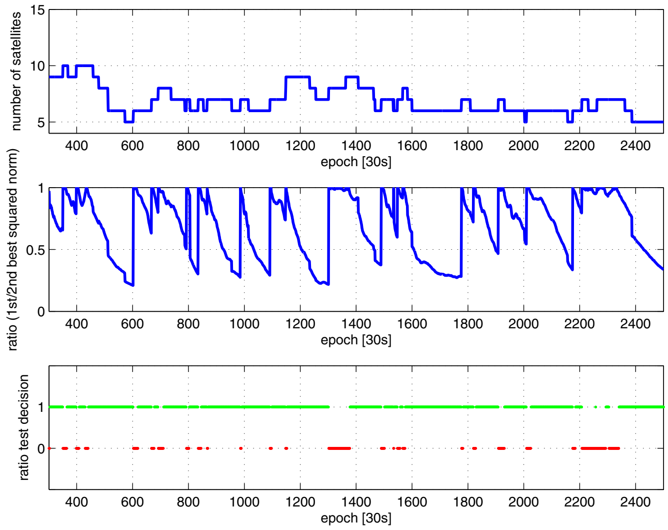

Figure 5 Number of satellites (top), ratio (middle) and outcome of FFRatio test (bottom; 1=passed, 0=failed) as function of the time span of 18.3h, for processing of the CORS-network consisting of 5 stations of the Southern California Integrated GPS network, **with no convergence time set** when a new satellite rises. The percentage of rejected FFRatio tests is 20%.

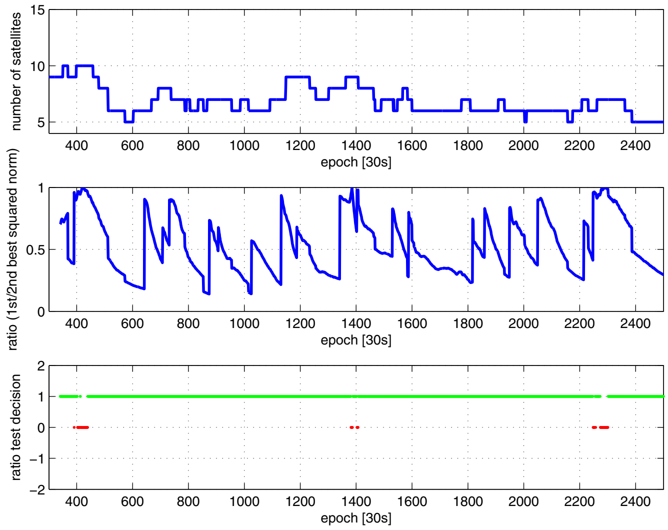

Figure 6 Number of satellites (top), ratio (middle) and outcome of FFRatio test (bottom; 1=passed, 0=failed) as function of the time span of 18.3h, for processing of the CORS-network consisting of 5 SCIGN stations, **with the convergence time fixed to 20 minutes** when a new satellite rises. The percentage of rejected FFRatio tests is 3%.

From Figure 5 it becomes also clear that if the Ratio Test is based on a *fixed* threshold of say ½ instead of a model-driven threshold as in the FFRatio Test, the time before a new satellite is included in the CORS Network ambiguity resolution would be significantly longer. For the current example about 90% of the Ratio Tests (with a fixed threshold of ½) are only passed after a 50-minute convergence time.

IV. Speeding Up CORS Network AR By Adding Information on the DD Ionospheric Delays

In this section we will focus on the reduction of the convergence times of CORS Network RTK ambiguity resolution, by incorporating information of the DD ionospheric delays to the model.

The ionosphere-float model (3) has been set up with the assumption that there is no a priori information on the DD ionospheric delays in a CORS Network. However, usually we have some information on the expected magnitude and smoothness of the DD ionospheric delays, depending on among others baseline length, location on Earth and time within the 11-yearly solar cycle, based on experience of the past. As example, Figure 7 shows the DD ionospheric delays based on fixed ambiguities of the CORS Network consisting of 5 stations, as described in Section III. The smoothness and baseline length dependence of the DD ionospheric delays can be clearly seen. Based on this experience, the a priori information for forthcoming epochs could be that the temporal behavior of the DD ionospheric delays is a random-walk stochastic process for which the size (for the longest baseline in the network) remains within ±2 m from zero.

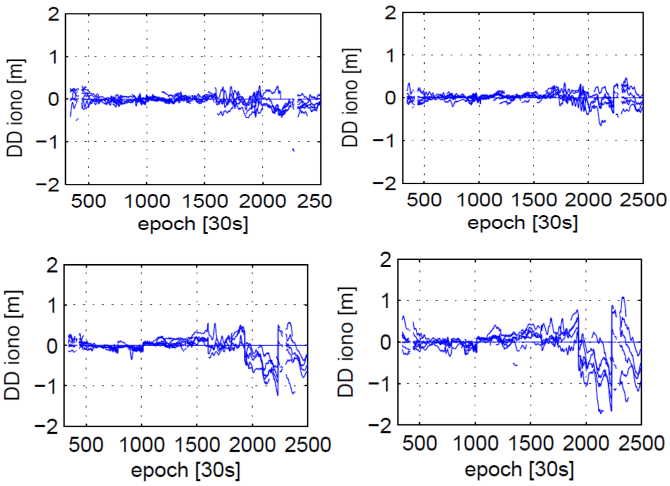

Figure 7 Ambiguity-fixed DD ionospheric delays for the 4 baselines within the simulated CORS Network consisting of 5 SCIGN stations. The GPS data applies to 11 April 2001. Top left: baseline LINJ-LVMS (89km); top right: baseline LINJ-CJMS (72km); bottom left: baseline LINJ-SPK1 (82 km); bottom right: baseline LINJ-TRAK (120km).

This a priori knowledge on the DD ionospheric delays can be added by using the *ionosphere-weighted* model for the processing of CORS Network RTK data. This model is an extension of the ionosphere-float model, by introducing stochastic pseudo-observables for the DD ionospheric delays:

$$E\left(\begin{pmatrix}\phi_1-\rho\\ \phi_2-\rho\\ p_1-\rho\\ p_2-\rho\\ \iota_1\end{pmatrix}\right)=\begin{pmatrix}M & -\mu_1 I & \lambda_1 I & \\ M & -\mu_2 I & & \lambda_2 I\\ M & \mu_1 I & & \\ M & \mu_2 I & & \\ & I & & \end{pmatrix}\begin{pmatrix}\tau_z\\ \iota_1\\ a_1\\ a_2\end{pmatrix} \tag{8}$$

The stochastic model is then extended as follows:

$$D\left(\begin{pmatrix}\phi_1-\rho\\ \phi_2-\rho\\ p_1-\rho\\ p_2-\rho\\ \iota_1\end{pmatrix}\right)=\begin{pmatrix}c_{\phi_1}^2 & & & & \\ & c_{\phi_2}^2 & & & \\ & & c_{p_1}^2 & & \\ & & & c_{p_2}^2 & \\ & & & & c_\iota^2\end{pmatrix}\otimes(D_n^T D_n)\otimes(D_m^T W_m D_m) \tag{9}$$

where c_ι^2 denotes the variance factor of the DD ionospheric pseudo-observations. For CORS Network RTK data processing the sample values of the DD ionospheric pseudo-observables are set to zero, while the choice of the ionospheric variance factor should depend on factors such as baseline length, location on Earth and progression of the solar cycle. This idea of *weighting* the ionospheric delays was first introduced by [11] and was successfully used to improve CORS-based RTK positioning for *users* in, among others, [7]. It is remarked that if $c_\iota^2=\infty$, the ionosphere-weighted model reduces to the ionosphere-float model.

To demonstrate the improvement in convergence time by using the ionosphere-weighted approach, Figure 8 shows the number of satellites, ambiguity ratio and outcome of the FFRatio Test of a 50-minute *ionosphere-float* processing of data of a network consisting of 3 CORS stations of the GPS Network Perth in Australia. The observations are collected on 15 February 2010. To get an impression of the ambiguity convergence times, both at the beginning of the session as well as when new satellites rise, the data are immediately included into the LAMBDA method (executed every epoch). From Figure 8 it follows that at the beginning the initialization time is about 6.3 min, while when new satellites rise, shortly before epoch 100 and before epoch 150, the necessary convergence times are 30 sec and 50 sec, respectively. During these time spans the FFRatio Test is not accepted. Next, consider the same dataset but now processed with the *ionosphere-weighted* model with c_ι = 10 cm, and a random-walk process to model the smoothness of the ionospheric delays in time, see Figure 9. With an undifferenced ionospheric standard deviation of 10 cm, the DD ionospheric standard deviation is 20 cm, and this implies that the magnitude of the DD ionospheric delays is allowed to vary within a bound of $\pm 3\sigma=\pm 60$ cm for 99.7% of the time. As can be seen from Figure 9, the convergence times both at the beginning and when the new satellite rise are much shorter than using the ionosphere-float approach. The convergence time at the beginning is reduced from 6.3 min to only 1 min, while the rising of the two satellites only takes 10 sec of convergence time against 30 and 50 sec using the ionosphere-float processing.

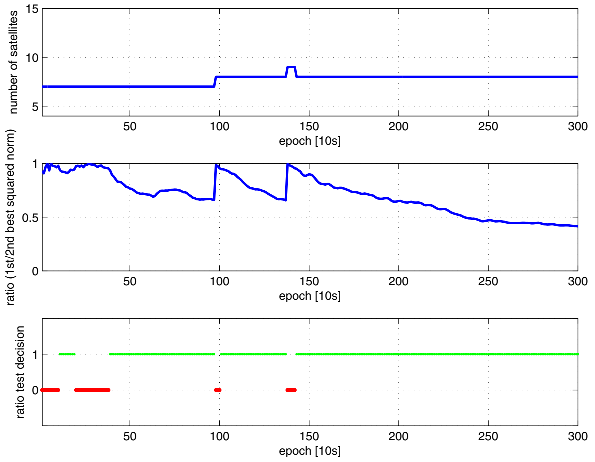

Figure 8 Number of satellites (top), ratio (middle) and outcome of FFRatio test (bottom; 1=passed, 0=failed) for a network consisting of 3 CORS stations of the GPS Network Perth, processed using the **ionosphere-float** model.

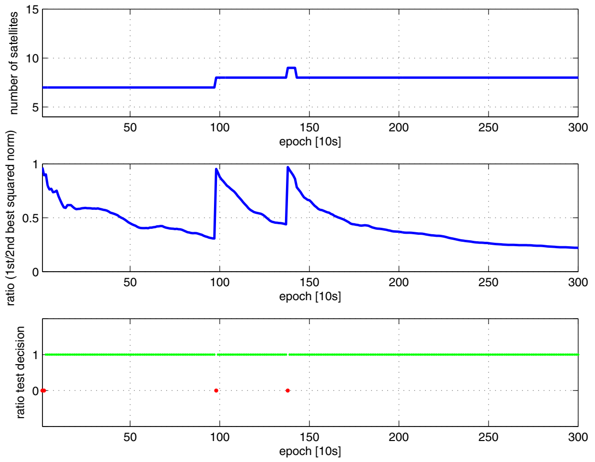

Figure 9 Number of satellites (top), ratio (middle) and outcome of FFRatio test (bottom; 1=passed, 0=failed) for a network consisting of 3 CORS stations of the GPS Network Perth, processed using the **ionosphere-weighted** model with an undifferenced ionospheric standard deviation of c_I = 10 cm.

V. CONCLUSIONS

Instantaneous (epoch-by-epoch) CORS Network RTK ambiguity resolution based on dual-frequency GPS is usually not feasible, preventing it from a true real-time service. However, in this paper it has been shown that the ambiguity convergence times, occurring at the beginning of operation, when a new satellite rises or after a failure (loss-of-lock), can be reduced when the CORS Network data processing is based on the following underlying assumptions. First, the ionosphere-weighted model should be used; incorporating the known positions of the CORS stations. The sample value of the DD ionospheric pseudo-observables are set to zero, but their standard deviation should be based on the CORS network size and the expected size of the DD ionospheric delays. Last, CORS Network RTK ambiguity resolution should consist of the LAMBDA method combined with the Fixed Failure-rate Ratio Test.

Further improvements which may speed up CORS Network RTK ambiguity resolution, but have not been investigated for the current paper, can be expected from *partial* ambiguity resolution. In this paper we tried to solve for the *full* CORS Network ambiguity vector in order to obtain the most precise ionospheric precision after ambiguity fixing, but it may be that the fixing of a subset of ambiguities could be done faster and still delivers ionospheric estimates of sufficient quality.

ACKNOWLEDGMENT

Professor Peter J.G. Teunissen is the recipient of an Australian Research Council (ARC) Federation Fellowship (project number FF0883188). This support is gratefully acknowledged.

REFERENCES

[1] G.R. Hu, H.S. Khoo, P.C. Goh and C.L. Law, "Development and assessment of GPS virtual reference stations for RTK positioning," Journal of Geodesy, Vol. 77, 2003, pp. 292-302.

[2] D. Odijk and P.J.G. Teunissen, "Speeding up CORS Network RTK ambiguity resolution," Proc. XXIV FIG International Congress 2010, Sydney, Australia, 11-16 April 2010, website: www.fig.net/pub/fig2010/techprog.htm

[3] P.J.G. Teunissen and D. Odijk, "Ambiguity Dilution Of Precision: Definition, properties and application," Proc. ION GPS 97, Kansas City, MO, 16-19 September 1997, pp. 891-899.

[4] B. Hofmann-Wellenhof, H. Lichtenegger and J. Collins, "Global Positioning System: Theory and Practice," 5th rev. ed., Springer, 2001.

[5] S.B. Bisnath and R.B. Langley, "Automated cycle-slip correction of dual-frequency kinematic GPS data," Proc. ION GPS 2000, Salt Lake City, UT, 19-22 September 2000, pp. 145-154.

[6] Y. Gao, Z. Li and J.F. McLellan, "Carrier phase based regional area differential GPS for decimeter-level positioning and navigation," Proc. ION GPS 97, Kansas City, MO, 16-19 September 1997, pp. 1305-1313.

[7] D. Odijk, "Weighting ionospheric corrections to improve fast GPS positioning over medium distances," Proc. ION GPS 2000, Salt Lake City, UT, 19-22 September 2000, pp. 1113-1123.

[8] P.J.G. Teunissen, "A new method for fast carrier phase ambiguity estimation," Proc. IEEE PLANS '94, Las Vegas, NV, 11-12 April 1994, pp. 862-873.

[9] P.J.G. Teunissen and S. Verhagen, "The GNSS ambiguity ratio-test revisited," Survey Review, Vol. 41, No. 312, 2009, pp. 138-151.

[10] Y. Zhang and C. Bartone, "A prototype long baseline DGPS concept demonstration for NDGPS high performance land applications," Proc. ION GNSS 2005, Long Beach, CA, 13-16 September 2005, pp. 362-375.

[11] B. Schaffrin and Y. Bock, "A unified scheme for processing GPS dual-band observations," Bulletin Géodesique, Vol. 62, No. 2, 1988, pp. 142-160.