GPS Solut (2016) 20:605–616
DOI 10.1007/s10291-016-0530-4

REVIEW ARTICLE

Snapshot positioning without initial information

Ignacio Fernández-Hernández[1,2] · **Kai Borre**[3,4]

Received: 25 July 2015 / Accepted: 11 March 2016 / Published online: 28 March 2016

Abstract Snapshot techniques are based on computing a position using only a set of digital signal samples captured over some milliseconds. Existing techniques require, in addition to the satellite ephemerides, a rough knowledge of the position and/or time at which the snapshot was captured. We propose a new method to instantaneously compute a snapshot position and time solution without any reference time or position. The method is based on the addition of a fifth unknown to the instantaneous Doppler equations, which accounts for a time difference between the reference time and the measurement time. Using this new system of equations at different initialization times separated by some hours, time uncertainties of days or weeks can be solved. The algorithm has been implemented in a snapshot GPS software receiver in MATLAB, proving that position accuracies of a few meters with time uncertainties of several weeks can be obtained in a few seconds.

Keywords Snapshot positioning · GNSS · Coarse-time navigation · Doppler positioning · Synchronization

✉ Ignacio Fernández-Hernández
Ignacio.fernandez-hernandez@ec.europa.eu

Kai Borre
borre@ssau.ru

[1] Aalborg University, Aalborg, Denmark

[2] BREY 7/156; EC DG GROW, 1049 Brussels, Belgium

[3] Samara State Aerospace University, Samara, Russia

[4] Vaarstvej 314, 9260 Gistrup, Denmark

Motivation

Thanks mainly to GPS, satellite navigation technologies have become ubiquitous. Miniaturized trackers attached to objects, animals, or human beings that are switched on sporadically at a given event or request may be widespread in the near future. For these applications, snapshot techniques are more suitable than standard receiver architectures. One could think, for example, of a low-power tracker attached to a migratory bird capturing several short signal snapshots during a period of several months. The tracker may wake up for example every few hours triggered by a timer. This timer is not a true real-time clock able to time-tag the snapshot. The tracker may also be switched on based on some other signal, such as temperature, pressure, vibration, moisture, speed, or acceleration. After several months, the tracker would download the snapshots to a server to reconstruct the seasonal movements. In this example, the server would have all the information about the satellite positions during the tracking period but would not have much information about when and where the snapshots were captured.

Snapshot positioning techniques are based on saving digital samples from the ADC (analog-to-digital converter) of a GNSS receiver front end and post-processing them to calculate a position and time solution. These techniques still require an initial time reference accurate to the level of a minute. We present here an algorithm for snapshot positioning techniques that allows computing a position and time solution without any initial position or timing information. It is based on a closed-form solution allowing calculating a position and time with an initial time error of more than an hour. Apart from its potential practical application, a main motivation for this work is to help solve the theoretical problem of how to calculate a radionavigation position with the minimum amount of information.

Background

A receiver has no synchronization with a signal just acquired, beyond the modulus of the spreading code length, e.g., 1 ms for GPS L1 C/A. It has to demodulate the signal data until a certain pattern is found, as is the case of TLM in the GPS signal. Only then, the receiver can get synchronized to the data bits and start interpreting the data. This includes decoding the satellite time reference called in GPS TOW (time of week) and WN (week number) first, and then the satellite ephemeris, which allow it to compute the satellite position and clock offset. This whole process may take between 30 s and 1 min for standard receivers, until a first position fix is obtained. The standard receiver acquisition process is described in detail in van Diggelen (2009) and Borre et al. (2007). Classic receiver architectures involving acquisition and tracking stages are described by A. J. Van Dierendonck in Chapter 8 of Parkinson and Spilker (1996) and by Chapters 11 and 12 of Misra and Enge (2010).

Snapshot techniques provide some advantages over standard acquisition and tracking positioning: they can provide an instantaneous position fix, they require much less power consumption, and the receiving device can be cheaper than an ordinary GNSS receiver. One of the earliest references found on snapshot techniques is Brown (1992). This reference presents a tracking sensor which periodically stores a snapshot of GPS signals, compresses it, and transmits it to a central workstation. As opposed to current snapshot techniques, it requires continuous snapshots every 0.5–5 s to obtain a few-meter accuracy, while modern techniques can obtain such an accuracy from a one-shot position solution through coarse-time navigation algorithms as described below. A one-shot high-sensitivity receiver is presented and characterized in Seco-Granados, et al. (2012). Dötterböck and Eissfeller (2009) describe a GPS/Galileo snapshot receiver and the signal detection theory behind it, showing some simulated performance, as also does Badia-Solé and Iacobescu-Ioan (2010). Since snapshot positioning is a major building block of assisted GNSS, van Diggelen (2009) describes it at the levels of signal processing and PVT (position, velocity, and timing).

Code phase ambiguity resolution

A classic receiver can compute the pseudorange measurement by subtracting the transmission time, which is known from the demodulated signal data, from the reception time. However, snapshot techniques are intended for applications and situations where the receiver cannot wait until the signal is synchronized. What the receiver is able to extract from a snapshot is a code phase measurement, i.e., a fractional measurement of the full pseudorange, as illustrated in a simplified way in Fig. 1. In order to form the full pseudorange, the snapshot receiver needs to know the number of full PRN codes between the satellite and the receiver to solve the code phase integer ambiguity.

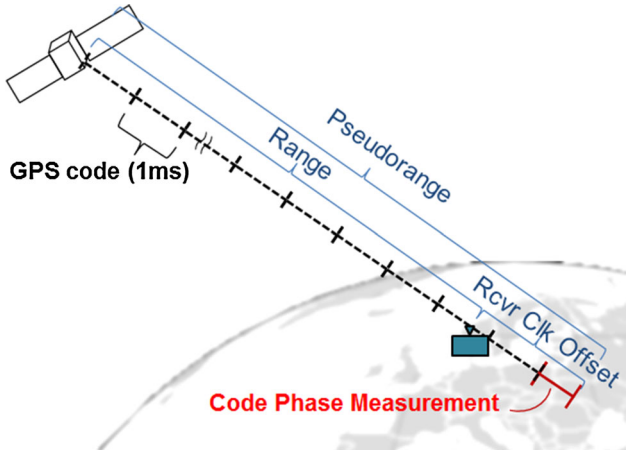

Fig. 1 Representation of the satellite-to-receiver pseudorange, as the true range plus the receiver clock offset, and as the sum of integer GPS 1-millisecond codes plus the code phase measurement

In order to solve the code phase ambiguity, the receiver needs to assign an integer number of codes N^i to the fractional code phase of each satellite i. Due to the receiver clock bias, which commonly affects all code phase measurements, even if the receiver and satellite positions at the code phase measurement time are roughly known, the receiver cannot know for sure what is the N^i of each satellite until it verifies the residuals of a solution. A method to resolve the code phase integer ambiguity to compute a full range measurement from an instantaneous fractional range measurement without integer rollovers is proposed in Chapter 5 of van Diggelen (2009). In order to solve the code phase ambiguity, an initial position and time accurate to around 100 km and a minute are necessary.

Coarse-time pseudorange navigation

Once the code phase ambiguities are solved and the pseudoranges are formed, the measurement time may be uncertain by some seconds, meaning that the satellite transmission times and positions are uncertain too. This problem can be solved by the coarse-time navigation algorithm (Peterson et al. 1995) explained in detail in van Diggelen (2009). This algorithm allows calculating a fix even if time is only known with an accuracy of a minute by adding a fifth unknown named "coarse time" to the navigation equations. The coarse-time navigation algorithm is based on the solution of the following equation:

$$\boldsymbol{\delta\rho} = \mathbf{H}\delta\mathbf{X} + \boldsymbol{\varepsilon} = \begin{bmatrix} -\mathbf{e}^i & 1 & \dot{\rho}^i \end{bmatrix} \begin{bmatrix} \delta x \\ \delta y \\ \delta z \\ \delta b \\ \delta tc \end{bmatrix} + \boldsymbol{\varepsilon} \quad (1)$$

where $\boldsymbol{\delta\rho}$ is the update of the a priori state, i.e., a column matrix with the difference between the predicted pseudoranges $\hat{\boldsymbol{\rho}}$ and the measured pseudoranges $\boldsymbol{\rho}$ ($\delta\rho = \boldsymbol{\rho} - \hat{\boldsymbol{\rho}}$), $\mathbf{H}$ is the observation matrix that relates the measurements with the state vector $\delta\mathbf{X}$, $-\mathbf{e}^i$ is the estimated receiver-to-satellite-i unit vector with the opposite sign, $\dot{\rho}^i$ is the pseudorange rate of satellite i, and $[\delta x\ \delta y\ \delta z\ \delta b\ \delta tc]^T$ is the update of the state vector $\delta\mathbf{X}$, or the vector of unknowns to solve, which includes the receiver position (x, y, z), the receiver bias b, and the coarse-time difference tc, i.e., the actual measurement time minus the estimated measurement time. The measurement and linearization errors are represented by $\boldsymbol{\varepsilon}$. Notice that $\dot{\rho}^i$ is often called range rate, avoiding the implication that it contains not only the range rate but also the satellite and receiver clock rates. The system of equations in (1) resembles that of standard PVT estimation as described by Misra and Enge (2010), Chapter 6, with the exception that it adds an unknown tc to the state vector to account for the coarse synchronization time error.

Instantaneous Doppler positioning

As mentioned before, in order to solve the code phase ambiguity, an initial position accurate to about 100 km is needed. Receivers connected to a communication network may easily obtain such an initial position. In other cases, van Diggelen (2009) describes in Chapter 8 the instantaneous Doppler algorithm, in which Doppler measurements are used to compute an initial position, accurate to some kilometers, sufficient to solve the code phase integer ambiguities and calculate the pseudoranges. For a static or slow-moving receiver, the instantaneous Doppler equations are:

$$\delta\mathbf{D} = \begin{bmatrix} \dot{\mathbf{e}}^i & 1 \end{bmatrix} \begin{bmatrix} \delta x \\ \delta y \\ \delta z \\ \delta fd \end{bmatrix} + \boldsymbol{\varepsilon} \quad (2)$$

where $\delta\mathbf{D}$ is the update of the a priori state, i.e., a column vector with the difference between the predicted Doppler measurements $\hat{\mathbf{D}}$ and the measured ones $\mathbf{D}$ ($\delta\mathbf{D} = \mathbf{D} - \hat{\mathbf{D}}$), $\dot{\mathbf{e}}^i$ is the derivative in time of $\mathbf{e}^i$, and $[\delta x\ \delta y\ \delta z\ \delta fd]^T$ is the update of the state vector, which estimates the receiver position and the frequency drift fd. Doppler and range rate have inverse signs, i.e., when the satellite is approaching, and thus the range is diminishing, the Doppler measurement is positive, and thus the frequency is higher:

$$\dot{\rho}^i = -\left(D^i - fd\right)\lambda + \varepsilon \quad (3)$$

where $\dot{\rho}^i$ is the range rate of satellite i, D^i is the Doppler measurement of the same satellite, fd is the receiver clock frequency drift, λ is the wavelength of the carrier frequency, and ε represents the errors in the measurement (e.g., frequency misalignment and receiver noise).

The instantaneous Doppler technique requires the knowledge of the initial time at which the Doppler measurements are estimated, accurate to the minute level. When the time is unknown, van Diggelen (2009) also presents a method to solve the time uncertainty by computing a solution at least every minute and determining the correct one by estimating the solution residuals. This method is further developed by Chen et al. (2014), requiring about 30-s processing in a standard processor to solve a 12-h time error. As proven below, the algorithm proposed in this article can solve a one-day ambiguity in less than half a second.

General description of the proposed algorithm

The proposed new algorithm is based on the combination of the instantaneous Doppler positioning algorithm and the coarse-time navigation algorithm, which leads to what we will call the coarse-time Doppler navigation algorithm. It introduces an additional unknown to the state vector of the instantaneous Doppler equations and relates it to the range rate measurements through the acceleration of the satellites as seen by the receiver.

Figure 2 depicts the case of one satellite and a receiver located at $\boldsymbol{P}$, having an initial time reference $t0$, off from the measurement time $t1$ by more than one hour. It also shows the estimated range rate $\dot{\rho}(t0)$ at time $t0$ and the measured range rate $\dot{\rho}(t1)$ at time $t1$ between the satellite and the receiver. It also depicts the estimated unit vector $\boldsymbol{e}$

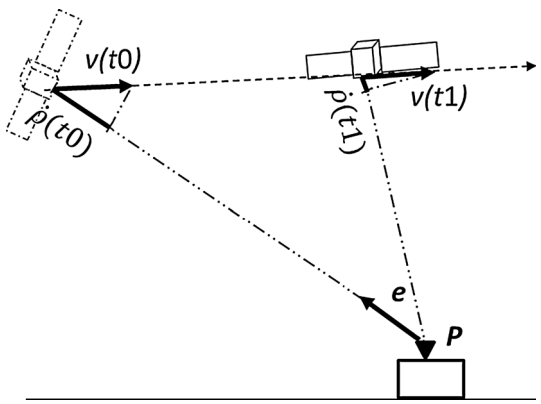

Fig. 2 Satellite range rate variation over time

pointing from the receiver to the estimated satellite position at $t0$.

Satellite-to-receiver range rate $\dot{\rho}(t0)$ at an instant $t0$ and $\dot{\rho}(t1)$ at an instant $t1$ differs by a magnitude that, in the proposed algorithm, can be approximated by the time increment between $t1$ and $t0$ multiplied by the satellite-to-receiver relative acceleration at $t0$, or differentiation in time of the range rate. We refer to this term as $\ddot{\rho}$. This acceleration relates the estimated range rate at $t0$ with the actual or measured range rate at $t1$ according to the equation

$$\dot{\rho}(t1) = \dot{\rho}(t0) + (t1 - t0) \cdot \ddot{\rho} \tag{4}$$

The equation assumes that $\ddot{\rho}$ is constant between $t1$ and $t0$, which is a sufficient assumption for the convergence to a solution if $t1$ and $t0$ differ by a few hours, depending on the case, as shown more in detail later. The limitations of this assumption are depicted in Figs. 3 and 4. Figure 3 shows the range rates of GPS satellites above 10° of elevation as seen by a receiver located at Aalborg, Denmark, while Fig. 4 shows the satellite–receiver accelerations. When a satellite is not visible, the range rates and accelerations appear as zero in both figures.

There is a linearization error because accelerations are not constant. However, this error does not prevent the convergence of the algorithm. The following sections show that, with GPS satellites, the closed-form solution here proposed always converged to the right solution for an initial error of at least 1.5 h.

Coarse-time Doppler navigation equations for a static receiver

For simplicity, we start with the description of the algorithm for a static or slow-moving receiver, up to some meters per second. In this case, the proposed state vector to be solved is

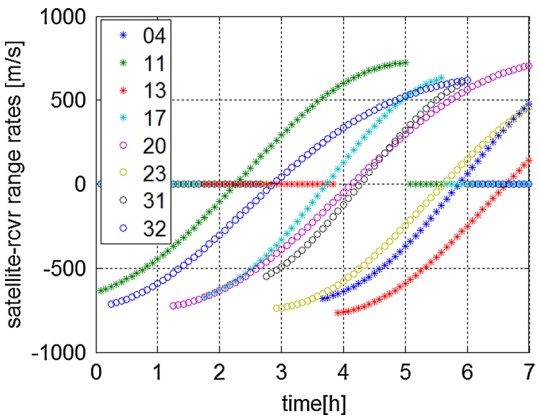

Fig. 3 Range rates of visible GPS satellites (PRN). March 2, 2010, 8:00–15:00 UTC, Danish GPS Centre, Aalborg, Denmark

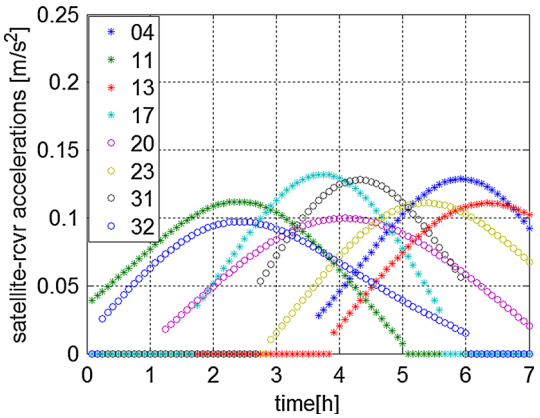

Fig. 4 Satellite–receiver accelerations of visible GPS satellites (PRN). March 2, 2010, 8:00–15:00 UTC, Danish GPS Centre, Aalborg, Denmark

$$\mathbf{X} = (x, y, z, fd, tc) \tag{5}$$

where x, y, and z are the receiver coordinates, fd is the receiver clock frequency drift, and tc is the time difference between the initial time $t0$ and the actual time $t1$. The system of equations to solve is:

$$\delta\mathbf{D} = \begin{bmatrix} \ddot{\mathbf{e}}^{\boldsymbol{i}} & 1 & -\ddot{\rho}^{i} \end{bmatrix} \begin{bmatrix} \delta x \\ \delta y \\ \delta z \\ \delta fd \\ \delta tc \end{bmatrix} + \boldsymbol{\varepsilon} \tag{6}$$

where $\delta\mathbf{D}$ corresponds to the vector of the differences between the Doppler measurements (m/s) and the Doppler estimation from the satellite data and a previous position and time, which for the first iteration is set to $P0$ at $t0$. The symbol $\ddot{\rho}^{i}$ denotes the satellite-i-to-receiver relative acceleration, and $\boldsymbol{\varepsilon}$ is the error associated with the measurements and the linearization process. This system of equations can be solved by a standard iterative method where the $\delta\mathbf{D}$ and $\delta x, \delta y, \delta z, \delta fd, \delta tc$ provide an update to the next iteration.

If the receiver is dynamic but has sensors that provide measurements of the receiver velocity, these measurements can be subtracted from the Doppler measurements and we can apply the equations for a static receiver. Velocity could be determined from an inertial unit, odometer, or any other source or signal. By having a static receiver or by not estimating the receiver velocity as part of the unknowns, the number of required satellites can be reduced to at least five or six if the measurement residuals are verified. If the frequency drift is known or very low, the number of required satellites can be reduced to at least four or five if residuals are verified. The same applies if the position height is estimated from another source, as, e.g., the geoid, or a more precise source.

Coarse-time Doppler navigation equations for a dynamic receiver

Before providing the equations for a dynamic receiver, we will mathematically derive the system of equations presented in (6) from the coarse-time pseudorange navigation Eq. (1), as follows:

$$\begin{aligned}\dot{\boldsymbol{\rho}} &= \dot{\mathbf{H}}\boldsymbol{\delta}\mathbf{X} + \mathbf{H}\dot{\boldsymbol{\delta}}\dot{\mathbf{X}} + \dot{\boldsymbol{\varepsilon}}\\ -\dot{\boldsymbol{\rho}} &= \begin{bmatrix} -\dot{\mathbf{e}}^{\boldsymbol{i}} & 0 & \ddot{\rho}^{i} \end{bmatrix} \begin{bmatrix} \delta x \\ \delta y \\ \delta z \\ \delta b \\ \delta tc \end{bmatrix} + \begin{bmatrix} -\mathbf{e}^{\boldsymbol{i}} & 1 & \dot{\rho}^{i} \end{bmatrix} \begin{bmatrix} \delta \dot{x} \\ \delta \dot{y} \\ \delta \dot{z} \\ \delta \dot{b} \\ \delta \dot{tc} \end{bmatrix} + \dot{\boldsymbol{\varepsilon}}\\ &= \begin{bmatrix} -\dot{\mathbf{e}}^{\boldsymbol{i}} & \ddot{\rho}^{i} \end{bmatrix} \begin{bmatrix} \delta x \\ \delta y \\ \delta z \\ \delta tc \end{bmatrix} + \begin{bmatrix} -\mathbf{e}^{\boldsymbol{i}} & 1 & \dot{\rho}^{i} \end{bmatrix} \begin{bmatrix} \delta vx \\ \delta vy \\ \delta vz \\ -\delta fd \\ 0 \end{bmatrix} + \dot{\boldsymbol{\varepsilon}} \end{aligned} \tag{7}$$

where the dot expresses the differentiation in time and $\delta vx, \delta vy$, and δvz express the receiver velocity. When the receiver is static or moving slowly, the relative velocity is zero, or close to zero, and therefore the system of equations becomes that in (6). When the receiver velocity needs to be taken into account, the system of equations is as follows:

$$\boldsymbol{\delta}\mathbf{D} = \begin{bmatrix} \dot{\mathbf{e}}^{\boldsymbol{i}} & \mathbf{e}^{\boldsymbol{i}} & 1 & -\ddot{\rho}^{i} \end{bmatrix} \begin{bmatrix} \delta x \\ \delta y \\ \delta z \\ \delta vx \\ \delta vy \\ \delta vz \\ \delta fd \\ \delta tc \end{bmatrix} + \dot{\boldsymbol{\varepsilon}} \tag{8}$$

where the state vector in (5) is extended to include the receiver velocity vx, vy, vz. The derivative $\dot{\mathbf{e}}^{\boldsymbol{i}}$ of the receiver-to-satellite unit vector for a given satellite at position $\vec{S}$ and a receiver at position $\overrightarrow{Rx}$, with a range ρ between the two, can be calculated as follows. The notation has been slightly modified here to distinguish between magnitudes ρ and vectors $\vec{\rho}$:

$$\begin{aligned}\dot{\vec{e}} &= \frac{\dot{\vec{\rho}}}{\rho}\\ \dot{\vec{e}} &= \left(\frac{1}{\rho^2}\right)\left[\dot{\vec{\rho}}\rho - \vec{\rho}\dot{\rho}\right]\\ &= \left(\frac{1}{\rho}\right)\left(\vec{V}_{s,rx} - \vec{e}\dot{\rho}\right)\end{aligned} \tag{9}$$

where $\vec{V}_{s,rx}$ is the vector of satellite velocity relative to the receiver. An equivalent expression is found in (8.6) of van Diggelen (2009).

The method therefore proposes the first closed-form solution allowing the calculation of a position and time with an initial time error of hours. For comparison purposes, Table 1 identifies and summarizes the closed-form solutions that have been developed over the past two decades on GNSS positioning with partial or no initial information.

General algorithm implementation including long-time uncertainty periods

The proposed algorithm can solve a timing uncertainty of more than 1 h by just solving the system of equations in (6) or (8). However, when the timing uncertainty is higher, the algorithm will not converge to the right solution, mainly due to the linearization errors of nonlinear equations. In order to eliminate the need for convergence over long periods, we calculate a solution for several initial times and choose the solutions with residuals below a given threshold. A way to implement this method is to split the time uncertainty interval into subintervals of a given duration, define an initial time $t0$ associated with each subinterval $T0_1$, $T0_2$, $T0_3$, $T0_4$, and $T0_5$ as per Fig. 5, until the full-time uncertainty interval is covered and, for each initial time, run the coarse-time Doppler navigation algorithm, which will provide, for each interval, a different time solution $T1_1$, $T1_2$, $T1_3$, $T1_4$, and $T1_5$, as per Fig. 5. If the method converges to a plausible solution for a certain interval, the obtained measurement residuals, in case of an overdetermined solution, will be below a certain threshold (THR1), as $T1_2$, $T1_3$, and $T1_5$ in the figure. We will call this method "cold snapshot." The approach followed is similar to that proposed by Chen et al. (2014), with the difference that,

Table 1 Closed-form solutions to GNSS positioning with partial or no initial information

	Initial time error of minutes	Correction of integer millisecond rollover	Initial position unknown	Initial time error of hours
Peterson et al. (1995)	X			
van Diggelen (2000, 2009)	X	X	X	
Fernández-Hernández (2015)	X	X	X	X

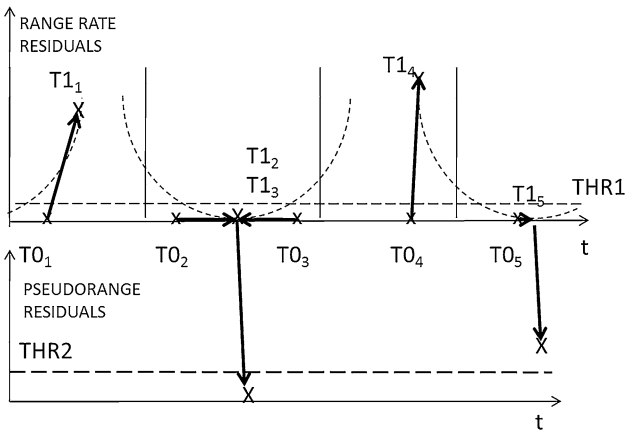

Fig. 5 Residual computation to determine the right "cold snapshot" solution. Residuals check for the coarse-time Doppler solution (*up*) and for the coarse-time pseudorange solution (*down*)

thanks to the additional unknown added to the state vector, there is one iteration every 3 h instead of every minute, which significantly reduces the processing time without any disadvantage to the knowledge of the authors. If needed, we can use the low-residual coarse-time Doppler navigation solutions to initialize a coarse-time pseudorange solution, which will converge to a correct and low-residual pseudorange solution below a certain residual threshold (THR2), if the initial position and time were roughly correct, as illustrated in Fig. 5.

The logic flow diagram for the extraction of coarse-time Doppler solutions is depicted in Fig. 6. As an outcome of the acquisition stage, measurements for each satellite of the satellite-to-receiver Doppler and range are obtained. Range rate measurements can be obtained from frequency Doppler measurements as well as code phase difference measurements or carrier phase difference measurements. As already mentioned, the algorithm requires a reference time $t0$ and position $P0$. The reference position $P0$ can be set to the center of the earth, or the center of the polygon formed by the satellite ground projection at $t0$ for the observed satellites. For a given position and time pair ($P0$, $t0$), the proposed method obtains and stores the receiver position, velocity, timing, and frequency offset in the following step by resolving (6) or (8), depending on the receiver dynamics.

The method then checks the plausibility of each solution ($P1$, $t1$) by comparing the residuals vector to a threshold THR1 in Fig. 5. If the solution is plausible, it is stored for later use and reported as an output of the method.

"Cold snapshot" with coarse-time pseudorange navigation

The "cold snapshot" method when based only on Doppler measurements will not yield an accuracy at the meter level

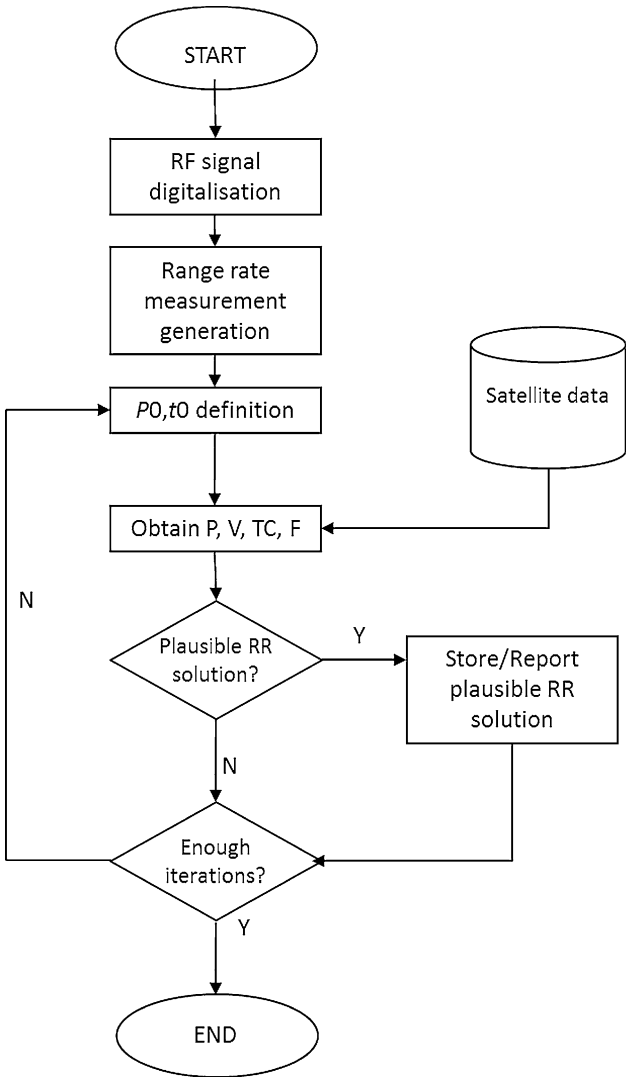

Fig. 6 Logic flow diagram of the "cold snapshot" algorithm

unless the signal carrier phases are stably tracked for some time. However, it can be used to calculate a more accurate instantaneous position and time based on range measurements. If a position with a higher accuracy needs to be obtained, the coarse-time pseudorange navigation method can be initialized with the coarse-time Doppler solution allowing an accuracy at the meter level, depending on the accuracy of the code phase measurements, as shown in Fig. 7.

Several Doppler solutions can be obtained and stored from a broad time uncertainty interval, as, e.g., those associated with $T1_2$, $T1_3$, and $T1_5$ in Fig. 5. In case a wrong Doppler solution such as $T1_5$ enters the coarse-time pseudorange navigation block, the pseudorange residuals check against THR2 in Fig. 5 will declare it as incorrect. The residual threshold in this case will be much lower than in the coarse-time Doppler case, commensurate with the

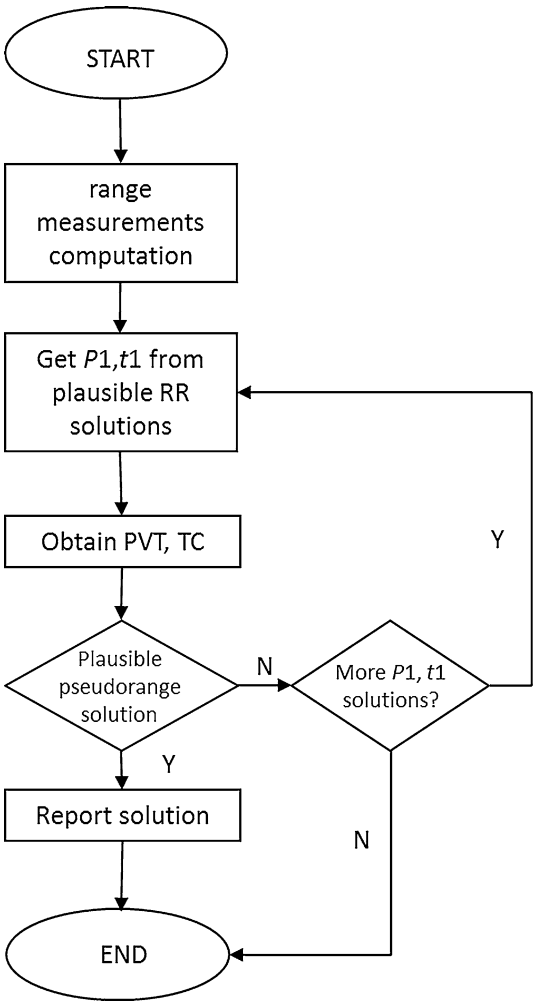

Fig. 7 Logic flow diagram of the "cold snapshot" algorithm combined with coarse-time pseudorange navigation

accuracy of the range-based solution. In the implementation presented, the PVT calculation has been performed by solving the system in (1) through least-squares estimation. Once reconstructed, the pseudoranges can be approximated as:

$$\rho = r + b + I + T + \varepsilon \tag{10}$$

where ρ is the pseudorange, r is the actual range, b is the receiver clock bias, I is the ionospheric delay, T is the tropospheric delay, and ε is the error due to other factors such as receiver noise, sampling and quantization, effects from the receiver clock drift, and multipath (Misra and Enge 2010). In the results later presented, the tropospheric error has been estimated following Goad and Goodman (1974), which describes the tropospheric correction function implemented in Borre et al. (2007). For the ionospheric correction, the Klobuchar (1987) model broadcast by GPS has been used. A Klobuchar error correction model has been coded in the MATLAB snapshot receiver for the experimental work of this research following the GPS interface specification (IS-GPS-200 2014).

Effect of GNSS orbital repeatability on Doppler navigation

If the time uncertainty interval is too broad, there will be periods of non-convergence, where the initial time ($T0_1$, $T0_4$), as per Fig. 5, does not allow the method to converge to a low-residual solution ($T1_1$, $T1_4$), and initial times ($T0_2$, $T0_3$, $T0_5$) where the method converges to a low-residual solution ($T1_2$, $T1_3$, $T1_5$). However, there may be low-residual Doppler solutions at wrong times ($T1_5$). This is due to the orbital repeatability of navigation satellites. Due to this effect, a low-residual Doppler solution can be obtained with a fixed periodicity. Given the 12-sidereal-hour orbital repeatability of GPS, this will always occur every 12 sidereal hours for GPS-only measurements: A low-residual solution is found at a position with approximately the same latitude but opposite longitude 12 sidereal hours later. The satellites will have completed their orbits and will be at the same positions with respect to an inertial reference frame, but the earth will have turned 180°. A low-residual solution at approximately the same latitude and longitude will be found 24 sidereal hours later, when the earth will have performed a full 360° turn. This 12-h ambiguity may pass the coarse-time Doppler residual threshold and can only be discarded at the later coarse-time pseudorange algorithm step. The application of ephemerides and clock corrections at incorrect times by several hours or days will lead to positioning errors that will be reflected into a solution with higher residuals, as illustrated in Fig. 5 and later in Fig. 10. Interestingly, thanks to the orbital imperfections and mainly due to clock drifts, the algorithm is able to work with any time uncertainties. This would not have been possible if the satellite orbits were perfect and the GPS atomic clocks had no drift because, for GPS only, the situation would repeat identically every orbit. Another way to solve this problem is to use measurements from at least two satellites, each from a different satellite constellation such as GPS, GLONASS, Galileo or Beidou, and with a different orbital period, to avoid the periodic repeatability of range rates from satellites from a single constellation. In this case, the repeatability period will correspond to the lowest common multiple of the orbital periods of the satellites used, which will be of several days.

Experimental results

We present some experimental results of the proposed algorithm. The results are based on real GPS snapshot data captures. A MATLAB-based snapshot receiver for GPS

Table 2 Summary of data capture properties and results

ID	Sampling frequency (MHz)	Intermediate frequency (MHz)	Location	Date and time (UTC)	Number and duration of snapshots (ms)	Avg. execution time. One-day time error (s)	RMS-2D, RMS-3D (m)
1	16.368	4.129945	Danish GPS Centre, Aalborg, Denmark	March 2, 2010 13:38	80, 10	0.46	5.6, 11.3
2	38.192	9.548	University of Boulder, Colorado, US	May 7, 2005, 19:10	100, 10	0.46	6.0, 8.2
3	16.368	4.129945	CTAE, Barcelona, Spain	March 25, 2010 15:14	80, 10	0.44	12.3, 35.2
4	16.368	4.129945	Gallecs, Barcelona, Spain	June 17, 2010 12:40	80, 10	0.32	6.8, 34,3
5	16.368	4.092	Universidad Autónoma de Barcelona, Spain	February 12, 2014 15:15	40, 10	0.41	6.4, 11.8

L1 C/A has been developed for testing the concept. It is partly based on Borre et al. (2007) and Badia-Solé and Iacobescu-Ioan (2010). Five data captures have been processed at different times and locations, as reported in Table 2. All data captures included GPS L1 C/A signals. The table presents the sampling and intermediate frequencies, the location and time, the number of snapshots calculated, and its duration. As results, the table presents the average execution time to solve a one-day time error, and the three-dimensional and two-dimensional RMS errors of the coarse-time range solutions initialized by the coarse-time Doppler solutions.

The average execution time shows that a one-day time error was solved in less than half a second, using a MATLAB non-optimized code in a standard Intel i7-3630QM CPU @ 2.4 GHz processor in Windows 8 at $\times 64$ bits. This means that, even before any translation to other languages such as C/C++ or any code optimization, the method can cover periods of days almost instantaneously. Software and hardware optimizations can decrease processing time to a quick resolution of time uncertainties of several weeks, months, or even years, allowing to virtually compute a PVT without any position and time reference.

The final column shows the accuracy of the code phase-based solution, which is in the order of that of standard GPS receivers. All solutions reported as correct were indeed correct, demonstrating the reliability of the method.

In the rest of the section, we will report in detail results based on data capture ID1 (Badia-Solé and Iacobescu-Ioan 2010), which is representative of the ensemble. The samples have been captured with a SiGe SE4110L GPS L1 integrated circuit (SiGe 2006). Here are the configuration parameters of the dataset for a one-month time error:

- True Position: Lat = 57.01473071°, Long = 9.9859041°, $h = 59.998$ m (WGS84)
- True Time: March 2, 2010, 12:14:04 UTC
- Receiver Conditions: static, open-sky
- Initial Position (X,Y,Z): (0,0,0)
- Initial Time: February 2, 2010, 02:00:00 UTC (28.43 days error)
- Coarse-Time Doppler Interval: 180 min
- Elevation Mask: 5°
- Integration Time for Each Solution: 10 ms
- Coherent Integration Time: 1 ms
- Non-Coherent Integrations: 10
- Doppler Solution—Residuals Vector Module Threshold (THR1): 15 m/s.
- Doppler Solution—Maximum Height: 20,000 m (above WGS84 geoid)
- Code Solution—Residuals Vector Module Threshold (THR2): 100 m
- Acquisition Method: code phase parallel acquisition
- Acquisition Frequency Bin Step: 500 Hz

The satellite ephemerides for the time uncertainty period are obtained from RINEX Navigation files (Gurtner 2007). Signal acquisition is performed coherently for 1 ms with 10 non-coherent integrations, for a total of 10 ms. The signal acquisition method used is based on the non-coherent code phase parallel acquisition as described in Borre et al. (2007). The Doppler and code phase measurements have been generated in open-loop mode, i.e., without tracking loops. One of the limitations of open-loop positioning is that, without any additional processing, the code phase accuracy is limited by the sampling frequency. If parallel code acquisition is performed, the acquisition peak will tell the sample with the highest peak, but the actual code delay will usually be between samples. In order to solve this, López-Risueño and Seco-Granados (2004) propose the interpolation between the samples around the peak. In the current implementation, a quadratic interpolation has been used. As regards to frequency accuracy, once the code phase measurement has been obtained, the signal can be de-spread to leave only the frequency carrier, and transformed through DFT (discrete Fourier transform) into a function of frequency, in which the maximum provides the Doppler measurement, with a higher accuracy

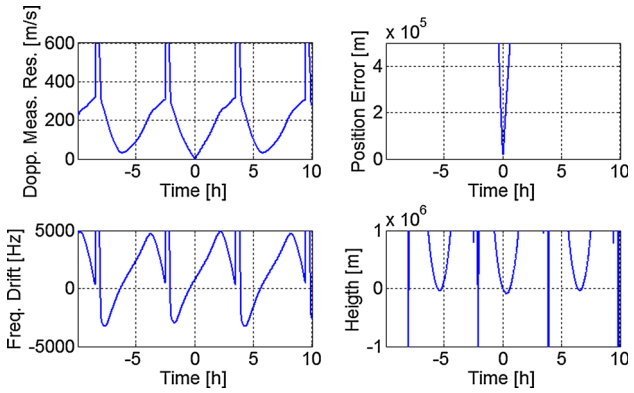

Fig. 8 Instantaneous Doppler positioning, ID1 (Danish GPS Center). Instantaneous Doppler solutions are calculated once every minute. *Top-left* Doppler measurement residuals vector module. *Top-right* Position error. *Bottom left* estimated frequency drift. *Bottom right* estimated position height

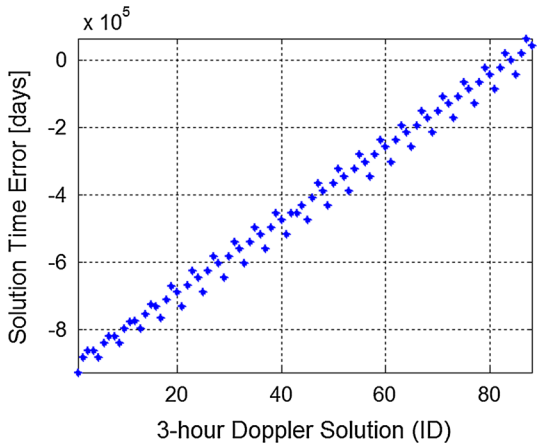

Fig. 9 Time error of 3-h coarse-time Doppler solutions over 10 days

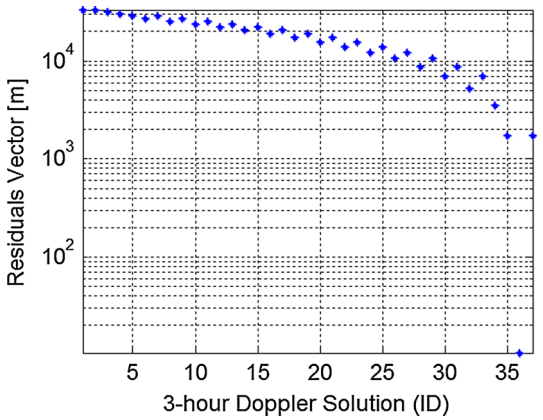

Fig. 10 Residual vector of plausible Doppler solutions over last 10 days

than that provided by the frequency bin resolution. Notice that the main goal of this research is not to maximize the snapshot accuracy but to provide sufficiently accurate measurements that allow validating the concept. We compute one coarse-time Doppler solution every 3 h. The proposed coarse-time Doppler equations in (6) have been solved using a Taylor first-order linearization, as in the standard literature, and a solution has been estimated through least squares (Misra and Enge 2010; Borre and Strang 2012).

In order to illustrate the behavior of Doppler-based solutions with real data, Fig. 8 presents the results of the standard instantaneous Doppler positioning as per (2) over a period of 20 h around the correct measurement time. Figure 8 top-left plot presents the module of the Doppler measurement residuals vector. It shows that the residuals are very low at the correct time (Time = 0). It also shows some relatively low residuals at ±360 min, i.e., every 6 h, or half the GPS 12-h orbital period, where the residual vector module is in the order of 100 m/s. The top-right plot shows the actual position error, which is very high in all cases except at the right timing. The bottom-left plot shows the estimated clock frequency drift. The bottom-right plot shows the height of the estimated position, which is close to the truth for the correct time and ±6 h.

In order to check the resolution of high time errors, the method was run over an initial time uncertainty of 1 month, for a total of 241 times. Out of them, only one solution had low enough code phase residuals and was considered the correct solution. Figures 9 and 10 show the results for the last 10 days.

Figure 9 shows the time error of each coarse-time Doppler solution. We can observe that the time error has a linear component and a periodic component, as every 12 h there are four solutions converging to a time error in a repeatable way. This can be expected as the Doppler solution residuals are repeatable every 12 h. Figure 10 shows, out of the 37 Doppler solutions that had low residuals, the residuals of the coarse-time pseudorange solution. We can see that there is only one solution (ID 36) with low enough residuals, the closest one having a residual vector module of around 2000 m, which seems high enough to be discarded.

The following figures show the sensitivity of the coarse-time Doppler solutions to errors in the Doppler measurements. Figures 11 and 12 show the time error and the position error, respectively, of the coarse-time Doppler solution in relation to the RMS error of the Doppler measurements for the 80 snapshots of Scenario ID1.

As expected, the figures show a correlation between the Doppler measurement quality and the position and time solution error. The method provides very good results with good-quality measurements, where the

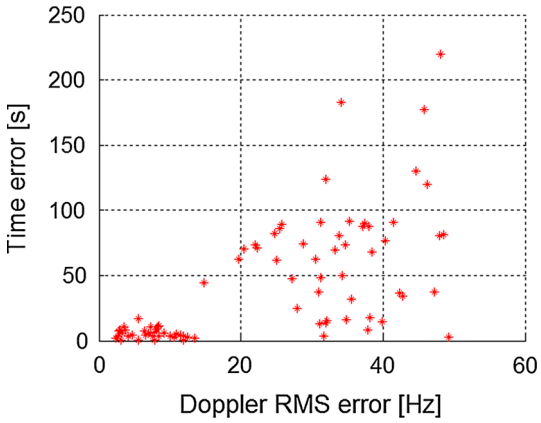

Fig. 11 Time error of the coarse-time Doppler solution versus Doppler measurement RMS error for 80 snapshots, ID1

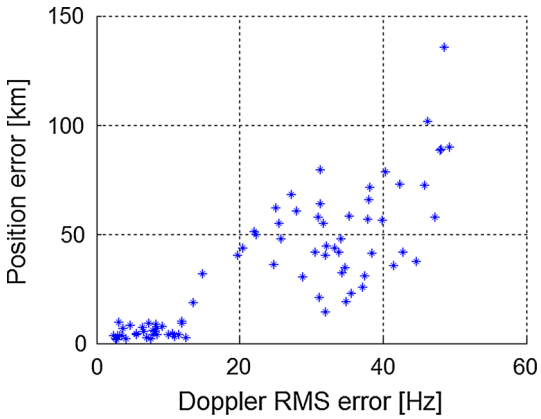

Fig. 12 Position error of the coarse-time Doppler solution versus Doppler measurement RMS error for 80 snapshots, ID1

Doppler RMS error is below 15 Hz (i.e., around 3 m/s), the position error in the order of 10–20 km, and the time error below 20 s. There is another region between 15 and 50 Hz of Doppler errors where the solution was less accurate but still generally allowing to solve the code phase ambiguity and calculate an accurate coarse-time pseudorange solution. However, there are six out of the 80 snapshots in this test where the measurement errors lead to position-time solutions with errors beyond 100 s that were discarded by the method at the pseudorange stage. Notice that the accuracy of the Doppler measurements is limited by the acquisition and low integration times used (1-ms coherent, 10 non-coherent integrations). Optimizing these parameters will lead to a higher availability. Notice that, in spite of the Doppler measurement errors, the method yielded no incorrect solution considered as correct, therefore proving full integrity and reliability not only with the ID1 80 snapshots but with the total of 380 snapshots analyzed.

Finally, Fig. 13 shows the 3D and 2D position errors for the 80 snapshots of Scenario ID 1, except those six for which the Doppler measurement quality did not allow the generation of a code-phase-based position. The RMS-2D error (root mean squared, two-dimensional horizontal) is around 5.6 m. The RMS-3D (root mean squared, three dimensional) error of is in the order of 11.3 m. The higher vertical than horizontal error is in accordance with expectations due to ionospheric effects and geometry. Figure 14 shows the obtained horizontal positions and the actual position in a map. The performance is not far from that of standard GPS receivers.

As the main purpose of the experimentation was to demonstrate the "cold snapshot" method based on the coarse-time Doppler navigation equations, a further characterization of the accuracy results is beyond the scope of this article. The accuracy obtained is considered in line with the expectations, taking into account that open-loop acquisition has been implemented, without any tracking loop fine-tuning or multipath mitigation strategy.

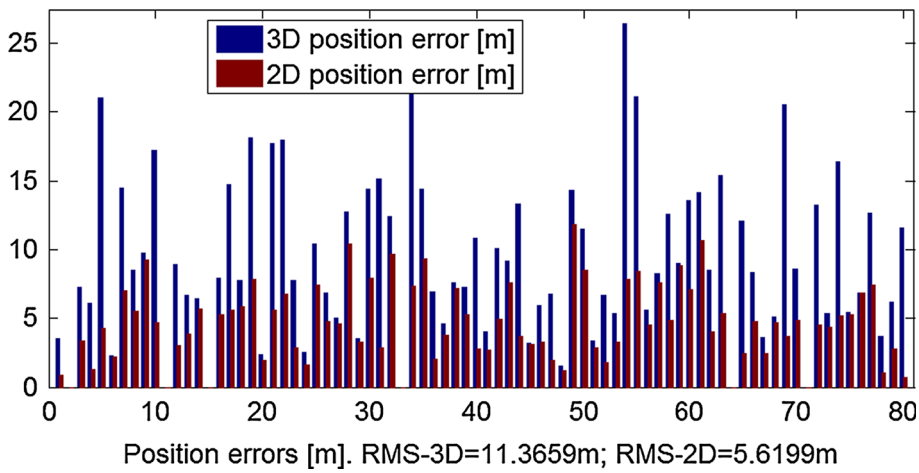

Fig. 13 Position errors of ID1 10-ms snapshots. "Cold snapshot" results—Danish GPS Centre, Aalborg, 2010

Fig. 14 Position errors of ID1. Horizontal "cold snapshot" results—Danish GPS Centre, Aalborg, 2010

Conclusions and further work

This article has presented a new algorithm to instantaneously calculate PVT over snapshots without any position and time reference. The algorithm is called coarse-time Doppler navigation and is based on the addition of a fifth state to the instantaneous Doppler equations (van Diggelen 2009), which accounts for the time difference, which can be in the order more than 1 h. The proposed system of equations can be obtained by time differentiation of the pseudorange coarse-time navigation equations. Time uncertainties up to days, weeks, or even months can be solved through several iterations at different times and the verification of the solution residuals. In principle, the proposed method can work with a snapshot taken virtually anywhere, at any time, provided that the measurements are of good-enough quality.

The method has been integrated in a MATLAB GPS L1 C/A snapshot receiver. Using the proposed coarse-time Doppler equations to provide an initial reference to coarse-time pseudorange equations, accuracies of few meters are obtained in less than half a second with time uncertainties of 1 day. Time uncertainties up to 1 month have been tested too.

Further work on this topic can cover the performance analysis of the algorithm using multi-GNSS measurements, the performance analysis of dynamic receivers under harsh reception conditions, and the fine-tuning of the measurement generation process.

Acknowledgments The authors would like to thank Oriol Badia Sole for granting access to the snapshot data captures from his Master's Thesis at Aalborg University, Prof. Gonzalo Seco-Granados, Prof. Torben Larsen, Prof. Dennis M. Akos, and the Journal anonymous reviewers for their comments. The work presented here is part of the research performed for a Ph.D. degree at Aalborg University. The information and views set out here are those of the authors and do not necessarily reflect the official opinion of any institution.

References

Badia-Solé O, Iacobescu-Ioan T (2010) GPS snapshot techniques. Aalborg University, Danish GPS Center

Borre K, Strang G (2012) Algorithms for global positioning. Wellesley-Cambridge Press, Wellesley

Borre K, Akos DM, Bertelsen N, Rinder P, Jensen SH (2007) A software-defined GPS and Galileo receiver: single-frequency approach. Birkhäuser, Basel

Brown, AK (1992) The TIDGET—a low cost GPS sensor for tracking applications. In: Proceedings of ION GPS 1992, Institute of Navigation. Albuquerque, NM, pp 661–669

Chen HW, Wang HS, Chiang YT, Chang FR (2014) A new coarse-time GPS positioning algorithm using combined Doppler and code-phase measurements. GPS Solut 18(4):541–551

Dötterböck D, Eissfeller B (2009). A GPS/Galileo software snap-shot receiver for mobile phones. In: Proceedings of the IAIN 2009 world congress. Stockholm, Sweden

Fernández-Hernández I (2015) Snapshot and authentication techniques for satellite navigation. Ph.D. Dissertation, Aalborg University, Faculty of Engineering and Science, Department of Electronic Systems

Goad CC, Goodman L (1974). A modified tropospheric refraction correction model. American Geophysical Union Annual Fall Meeting. San Francisco

Gurtner W (2007) RINEX: the receiver independent exchange format version 2.10. ftp://igscb.jpl.nasa.gov/igscb/data/format/rinex210.txt

IS-GPS-200 (2014) GPS interface specification. The US Government

Klobuchar J (1987) Ionospheric time-delay algorithms for single-frequency GPS users. IEEE Trans Aerosp Electron Syst 3:325–331

López-Risueño G, Seco-Granados G (2004) Measurement and processing of indoor GPS signals using a one-shot software receiver. 2nd ESA workshop on satellite Navigation User Equipment Technologies(NAVITEC 2004). Noordwijk, The Netherlands

Misra P, Enge P (2010) Global positioning system: signals, measurements and performance (revised second edition). Ganga-Jamuna Press, Lincoln

Parkinson BW, Spilker JJ (1996) Global positioning system: theory & applications, vol 1, 1st edn. American Institute of Aeronautics and Aeronautics, Reston

Peterson B, Hartnett R, Ottman G (1995). GPS receiver structures for the urban canyon. In: Proceedings of ION GPS 1995, Institute of Navigation, Palm Springs, CA, pp 1323–1332

Seco-Granados G, Lopez-Salcedo JA, Jiménez-Baños D, López-Risueño G (2012) Signal processing challenges in indoor GNSS. IEEE Signal Process Mag 29(2):108–131

SiGe SE4110L (2006) "SE4110L PointChargerTM GPS Receiver IC." 73-DST-01. SiGe Semiconductor Inc

van Diggelen F (2000) Method and apparatus for time-free processing of GPS signals. U.S. patent 6417801

van Diggelen F (2009) A-GPS: assisted GPS, GNSS, and SBAS. Artech House, Norwood

Ignacio Fernández-Hernández is an engineer from ICAI (Madrid); has an MBA from LBS, London; and recently obtained a Ph.D. in Electronic Systems from Aalborg University for his research on snapshot algorithms and GNSS authentication. He works at the European Commission on the definition of Galileo services.

Kai Borre originally graduated as a chartered surveyor. Kai Borre obtained his Ph.D. in geodesy from Copenhagen University and a Doctor of Technology degree from Graz University of Technology. He has been a full professor in geodesy at Aalborg University since 1976. Borre is a co-author of the widely used textbooks, *Linear Algebra, Geodesy, and GPS* and *A Software-Defined GPS and Galileo Receiver; Single-Frequency Approach.* He currently works as a Senior Researcher at the Samara State Aerospace University, Russia.