Multisatellite Tracking GNSS Receivers in Multipath Environments

Kaspar Giger, *Technische Universität München, Germany*
Christoph Günther, *Technische Universität München, and German Aerospace Center (DLR), Germany*

BIOGRAPHY

Kaspar Giger received the M.S. degree in electrical engineering and information technology from the Swiss Federal Institute of Technology (ETH), Zurich, Switzerland, in 2006. In his M.S. thesis, he worked on an implementation and optimization of a real-time MPEG4/AVC H.264 encoder on a TI DM642 DSP platform. He is currently pursuing the Ph.D. degree at the Institute for Communications and Navigation, Technische Universität München, Munich, Germany, working on new signal tracking algorithms.

Christoph Günther studied theoretical physics at the Swiss Federal Institute of Technology (ETH), Zurich, Switzerland. He received his diploma in 1979 and completed his PhD in 1984. He worked on communication and information theory at Brown Boveri and Ascom Tech. From 1995, he led the development of mobile phones for GSM and later dual mode GSM/Satellite phones at Ascom. In 1999, he became head of the research department of Ericsson in Nuremberg. Since 2003, he is the director of the Institute of Communications and Navigation at the German Aerospace Center (DLR) and since December 2004, he additionally holds a Chair at the Technische Universität München, Munich, Germany. His research interests are in satellite navigation, communication, and signal processing.

ABSTRACT

In global navigation satellite systems (GNSS) multipath propagation potentially leads to performance degradation and a reduced robustness of the receivers. Existing algorithms aiming at reducing the impact of multipath have an increased computational complexity as they typically require at least additional correlators. On the other hand multisatellite tracking algorithms provide a high robustness in fading environments. But their behavior in multipath is not documented yet. Therefore this paper aims at comparing three different signal tracking approaches in a simulated realistic multipath environment: state-of-the-art scalar tracking loops, vector delay locked loops and joint tracking loops. The performance results show the benefit of the carrier usage in the tracking. In both, the scalar tracking loops and the multisatellite tracking loops. Thus the joint tracking receivers prove to be a robust and yet precise tool for standalone position estimation with GNSS signals.

1 INTRODUCTION

Among the sources of error in GNSS multipath propagation is one of the largest contributions. It is therefore important to know how receivers perform in such an environment and how to fight the multipath. Various techniques are documented in literature to reduce or even eliminate multipath errors. They range from signal design [1] to antenna design [2, 3] and receiver algorithms. The receiver algorithms can be divided into two classes: multipath reduction and multipath estimation. The first ones are typically less computationally intensive as they simply try to setup the correlators and discriminators in a way that the multipath error envelope is as small as possible. This comprises amongst others the narrow correlator [4], strobe correlators [5] and further approaches as summarized in [6]. Techniques trying to estimate and thus eliminate the multipath bias are typically maximum likelihood based. The maximization of the likelihood function can be solved either by parallel code-tracking loops [7], a multitude of correlators [8] or more sophisticated algorithms like the sequential Monte Carlo filter [9]. Common to practically all advanced multipath reduction and mitigation algorithms is the need for additional correlators. As this is typically expensive for low-cost receivers, this paper only considers receiver algorithms working with the existing set of correlators (one early, late and prompt correlator per channel).

Besides the superposition of the line-of-sight signal with echoes, fading and shadowing is also likely to happen in multipath-affected environments. To cope with short signal outages of a subset of the tracked satellites, multisatellite algorithms have been shown to be a viable solution [10]. A differentiation between noncoherent and coherent algorithms can be made. The older noncoherent methods originate from the works of Copps et al. [11] and Spilker (vector delay locked loop, *VDLL*) [12]. It was shown in [13] that the VDLL has a reduced risk of losing lock and thus an enhanced robustness. A coherent multisatellite (and multifrequency) algorithm was introduced by Giger et al. (joint tracking loop, *JTL*) [14]. It also exploits the spatial (and spectral) correlation between the received signals and can in this way reduce the probability of cycle slips in the carrier-phase tracking. It was shown in [15] that this receiver structure can also improve the positioning (position domain joint tracking loop, *PTL*). Although all multisatellite algorithms have shown an enhanced robustness compared to standard receivers, their performance in multipath scenarios is not yet known.

Therefore the objective of this paper is to analyze and quantify the performance of multisatellite tracking algorithms in realistic multipath scenarios. The paper is organized as follows: section 2 details the simulation environment and channel model used. In section 3 an overview over the considered receiver types is given, followed by the simulation results in section 4 and their discussion in section 5. The paper is finally concluded in section 6.

2 SOFTWARE CONSTELLATION SIMULATOR

To allow a quantification of the positioning accuracy of any receiver algorithm, the receiver position and clock bias must be precisely known. Especially in the considered multipath environments such information is difficult to obtain from real measurements. Therefore a software constellation simulator in combination with a realistic GNSS channel model is used. This section details the generation of the simulated signal samples at the receiver.

2.1 Receiver and Satellite Trajectory

In a first step the space-time trajectory of the receiver is generated, i.e. its position $\vec{r}\,(t)$ and clock-offset $c\delta t(t)$ over time. Either an interpolated discrete-time random-walk process is simulated or a polynomial model applied. For the random-walk process a trajectory in the state-space is generated at a low temporal resolution and interpolated subsequently. The state-space is defined as the receiver position and time and the respective derivatives:

$$\xi(t) = \begin{pmatrix} \vec{r}(t) \\ c\delta t(t) \end{pmatrix},$$
$$\partial\xi(t) = \left(\xi^T(t), \quad \dot{\xi}^T(t), \quad \ddot{\xi}^T(t), \quad \ldots\right)^T.$$

Above the matrix transpose is written as $(.)^T$. The state-space description models the temporal evolution of position and clock-offset:

$$\partial\xi(t_{i+1}) = \Phi_\xi \partial\xi(t_i) + C_Q w(t_i).$$

The state-space transition matrix Φ_ξ can be found to read

$$\begin{aligned} \Phi_\xi &= \Phi_n \otimes I_4, \\ \text{with } \Phi_n &= \begin{pmatrix} 1 & T & \ldots & T^{(n-1)}/(n-1)! \\ 0 & 1 & \ldots & T^{(n-2)}/(n-2)! \\ \vdots & & \ddots & \vdots \\ 0 & 0 & \ldots & 1 \end{pmatrix}, \\ T &= t_{i+1} - t_i. \end{aligned} \tag{1}$$

and $\otimes$ the Kronecker product. The product $C_Q w(t_i)$ models the randomization due to the random process noise. The matrix C_Q can be found by applying the Cholesky decomposition to the (discrete-time) process noise covariance matrix Q. The vectors $w(.)$ are drawn randomly from a multivariate Gaussian distribution according to

$$\mathcal{E}\left[w(t_i)\right] = 0, \; \mathcal{E}\left[w(t_i) w^T(t_j)\right] = I \cdot \delta_{i,j},$$

where $\delta_{i,j}$ denotes the Kronecker delta and I an identity matrix of appropriate dimensions. The expectation is written as $\mathcal{E}\,[.]$.

The trajectory of the satellites, i.e. $\vec{r}^{\,k}(t)$, $c\delta t^k(t)$, follow their description in the broadcast ephemerides.

2.2 Line-of-sight Signal

The propagation delay τ^k of the signal of the k-th satellite can be found by solving the following equation, where it's assumed that all coordinates are given in the same Euclidean frame[1]:

$$c\tau^k(t) = \left\| \vec{r}\,(t) - \vec{r}^{\,k}\left(t - \tau^k(t)\right) \right\|,$$

with c the speed of light. Consequently the received signal at time t_{Rx} (measured by the receiver's clock) now reads

$$r^k(t_{\text{Rx}}) = s^k\Big(t_{\text{Rx}} - \tau^k(t_{\text{Rx}} - \delta t) - \delta t + \delta t^k\Big), \tag{2}$$

with δt and δt^k the clock offset of the receiver and satellite k at the time of reception and transmission respectively, and $s^k(t_{\text{Tx}})$ the signal transmitted by the satellite at time t_{Tx} (where t_{Tx} is measured by the satellite's own clock). The transmitted signal consists of the binary navigation bits $b^k(.)$, spread by the code $c^k(.)$ and modulated onto the carrier frequency ω_c:

$$s^k(t) = A^k b^k(t) c^k(t) \cos(\omega_c t). \tag{3}$$

[1] In reality the propagation delay would also contain the atmospheric delays. These are omitted here to allow the study of only the multipath performance of the receivers.

Above the amplitude of the transmitted signal is denoted by A^k and is assumed to be constant. For the simulations, random binary navigation bits b^k were chosen randomly, with plus and minus one equiprobable. The code- and carrier-phase of the received signal can now easily be found by plugging Eq. (3) into Eq. (2).

2.3 Multipath Propagation

In the simulation of the multipath environment, the Land Mobile Satellite Channel Model of the German Aerospace Center (DLR) is used. Details of the model can be found in [16]. Given a choice of parameters statistically describing an artificial scenery, the environment is created. It consists of a road the receiver is moving on, house fronts besides the road, as well as light poles and trees. An exemplary realization is illustrated in Fig. 1.

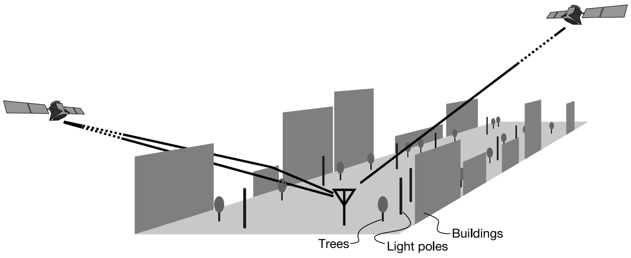


Fig. 1. Exemplary multipath scenery.

The environment together with the user's motion and the direction to the satellites defines the channel impulse response for a certain time instant. The channel impulse response is given by a number of line-of-sight paths and echoes (in total N^k paths). They are described by the excess delay, τ_i, phase rotation, φ_i and amplitude scaling, a_i. In complex baseband notation, the time-varying impulse response for satellite k reads

$$h^k(t,\tau) = \sum_{i=1}^{N^k(t)} a_i^k(t)\delta\left(\tau - \tau_i^k(t)\right) e^{j\varphi_i^k(t)}. \qquad (4)$$

Finally, the received signal can be written out by plugging Eq. (4) and (2) into Eq. (3). It has to be taken into account that the delays output by the channel model are w.r.t. the line-of-sight signal, i.e. a path with $\tau_i^k = 0$ is still delayed as described by Eq. (2).

2.4 Frontend Processing

The software constellation simulator has to model all components between signal generation and the sampled and quantized signal at reception. Therefore the received satellite signals are oversampled at the intermediate frequency, at a sampling frequency fulfilling

$$f_s > 2f_i + 2B,$$

where f_i is the intermediate frequency and B the (one-sided) bandwidth of the generated spreading code. The above condition avoids aliasing due to the sampling. After the sampling at the intermediate frequency a discrete-time bandpass filter is applied corresponding to the analog frontend-filter in a real receiver, $H_f(e^{j\theta})$.

Finally the samples are scaled by the AGC (according to the optimal ratio between noise standard deviation and maximum quantization level, e.g. [17]) and quantized to a desired number of bits. The quantization output r_q can be defined as a function of the input sample r_f and the number of bits used, N_b [18]:

$$r_q(r_f) = \begin{cases} \left(2^{N_b} - 1\right)/2, & \text{if } r_f \geq 2^{N_b-1} - 1 \\ (k - 1/2)\,, & \text{if } k > r_f \geq k - 1 \\ \left(1 - 2^{N_b}\right)/2, & \text{if } 1 - 2^{N_b-1} > r_f \end{cases}$$
$$\text{where } k \in \left\{2 - 2^{N_b-1}, \ldots, 2^{N_b-1} - 1\right\}.$$

The complete frontend chain is illustrated in Fig. 2.

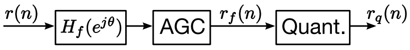


Fig. 2. Modeled frontend processing.

3 RECEIVERS

This section details the receivers considered in the performance analysis of this paper. As a side-constraint, only those receiver algorithms should be considered that operate with a comparable computational complexity like a state-of-the-art receiver. In other words, techniques requiring additional correlators, such as e.g. the MEDLL [8], sequential Monte Carlo filters [9] or the like, are not taken into account in this study. In this way it would be possible to run the discussed algorithms on a commercial receiver by just updating the firmware.

The following four receiver algorithms are considered:

- State-of-the-art tracking (PLL + DLL) with Kalman filter positioning
- State-of-the-art tracking (PLL + carrier-aided DLL) with Kalman filter positioning
- Vector delay locked loop (VDLL)
- Position domain joint tracking loop (PTL)

3.1 Scalar Tracking

The description of a receiver with a scalar tracking module can be found in any textbook, e.g. [19]. For every received signal, an independent tracking loop is setup. It consists of a phase locked loop (PLL), tracking the carrier-phase of the received signal, and a delay locked loop (DLL), tracking the code-phase of the received signal. Optionally, the code-phase tracking can be aided by the PLL, since spreading code and carrier share the same line-of-sight dynamics. A scaling factor $\alpha = 2\pi f_{\text{code}}/\omega_c$ has to be further applied to account for the different wavelength of code and carrier. The code-frequency (or chipping rate) f_{code} is given in chips per second and the carrier-frequency ω_c in radians per second. The block diagram of one channel is illustrated in Fig. 3.

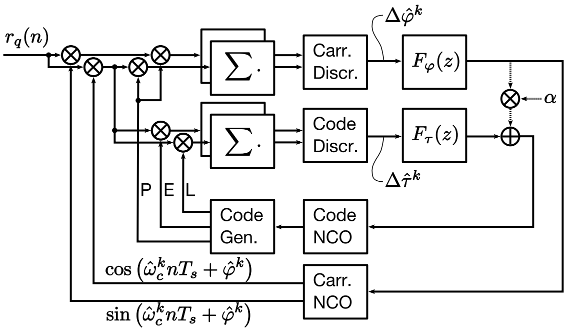


Fig. 3. Scalar spread spectrum signal tracking.

The pseudorange measurements obtained by tracking the code-phase are then used for the positioning. To allow a fair comparison to the multisatellite tracking algorithms, a Kalman filter is used for positioning in this configuration. The state-space process model is equal to the one of the vector delay locked loop, for which the reader is referred to (section 3.2). The corrected pseudoranges, $\tilde{\rho}_t^k$, are used as measurements in the filter:

$$y_{\mathrm{sc.},t} = \begin{pmatrix} \tilde{\rho}_t^1 \\ \vdots \\ \tilde{\rho}_t^K \end{pmatrix} = \begin{pmatrix} \rho_t^1 + \vec{e}_t^{\,1} \cdot \vec{r}_t^{\,1} + c\delta t^1 \\ \vdots \\ \rho_t^K + \vec{e}_t^{\,K} \cdot \vec{r}_t^{\,K} + c\delta t^K \end{pmatrix},$$

where K denotes the number of tracked satellites, t the discrete time-step and $\vec{e}^{\,k}$ the unit vector pointing from satellite k to the receiver. The dot product between two vectors is represented by the dot, i.e. $a \cdot b = a^T b$. Consequently the measurement matrix used in the filter consists of the unit vectors and a column of ones:

$$y_{\mathrm{sc.},t} = H_{\mathrm{sc.},t} x_{\mathrm{sc.},t} + v_{\mathrm{sc.},t},$$
$$H_{\mathrm{sc.},t} = \begin{pmatrix} \left(\vec{e}_t^{\,1}\right)^T & 1 & 0 & \ldots & 0 \\ \vdots & \vdots & \vdots & & \vdots \\ \left(\vec{e}_t^{\,K}\right)^T & 1 & 0 & \ldots & 0 \end{pmatrix}.$$

The measurement noise covariance is modeled as diagonal, with the diagonal entries determined by the closed-loop code-phase tracking jitter, e.g. [20]

$$[R_{\mathrm{sc.}}]_{k,k} = \frac{B_L\left(1 - B_L T_i/2\right) d}{2C^k/N_0} \left(\frac{c}{f_{\text{code}}}\right)^2 \cdot \left(1 + \frac{2}{T_i C^k/N_0(2-d)}\right),$$

with C^k/N_0 the measured carrier-to-noise density ratio of the k-th channel, B_L the bandwidth of the DLL, T_i the predetection correlation interval duration and d the Early-Late correlator spacing in chips.

3.2 Vector Delay Locked Loop (VDLL)

The vector delay locked loop follows the description in [12]. The underlying Kalman filter estimates the receiver position $\vec{r}_t$ and clock offset δt_t and their derivatives. For simplicity the vector ξ_t was introduced earlier, i.e. $\xi_t = \left(\vec{r}_t^{\,T},\ c\delta t_t\right)^T$. The state-vector can now be written as

$$x_{\mathrm{VDLL},t} = \partial\xi_t = \left(\xi_t^T,\quad \dot{\xi}_t^T,\quad \ddot{\xi}_t^T \quad \ldots\right)^T.$$

Using the above definition, the state-transition equation can easily be found to read

$$\begin{aligned} x_{\mathrm{VDLL},t+1} &= \Phi_{\mathrm{VDLL}} x_{\mathrm{VDLL},t} + w_{\mathrm{VDLL},t} \\ &= \left(\Phi_n \otimes I_4\right) x_{\mathrm{VDLL},t} + w_{\mathrm{VDLL},t}, \\ \text{where } \mathcal{E}&\left[w_{\mathrm{VDLL},t} w_{\mathrm{VDLL},t'}^T\right] = Q_{\mathrm{VDLL}} \cdot \delta_{t,t'}. \end{aligned}$$

The matrix Φ_n was already defined in Eq. (1). The identity matrix of dimension $n \times n$ is denoted by I_n. If the code-phase and -frequency as well as the carrier-frequency of the local replica are computed according to the state estimate $\hat{x}_{\mathrm{VDLL}}$, then the code discriminator outputs approximately the difference between the true code-phase, τ, and the estimated code-phase $\hat{\tau}$, i.e.

$$c\tau_t^k = c\tau^k(x_{\mathrm{VDLL},t}) = \left\|\vec{r}_t - \vec{r}_t^{\,k}\right\| + c\left(\delta t_t - \delta t_t^k\right), \quad (5)$$

$$c\hat{\tau}_t^k = c\tau^k(\hat{x}_{\mathrm{VDLL},t}) = \left\|\hat{\vec{r}}_t - \vec{r}_t^{\,k}\right\| + c\left(\hat{\delta t}_t - \delta t_t^k\right), \quad (6)$$

$$\begin{aligned} D_\tau^k &= D_\tau\left(c\tau_t^k - c\hat{\tau}_t^k\right) \approx c\tau_t^k - c\hat{\tau}_t^k + \eta_t^k \\ &\approx \vec{e}_t^{\,k} \cdot \left(\vec{r}_t - \hat{\vec{r}}_t\right) + c\left(\delta t_t - \hat{\delta t}_t\right) + \eta_t^k, \end{aligned} \quad (7)$$

where η_t^k is the noise contained in the discriminator output. Note that a noncoherent discriminator needs to be used here.

A block diagram for this receiver is shown in Fig. 4. For convenience the carrier downmixing is represented simply by a multiplication with the complex exponential. Therefore the correlation results are complex and can directly be used as input into the noncoherent code-phase discriminator.

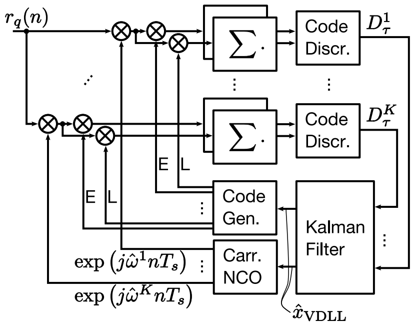

$r_q(n)$
$\sum \cdot$
Code Discr.
D_τ^1
D_τ^K
E
L
Code Gen.
Kalman Filter
Carr. NCO
$\exp\left(j\hat{\omega}^1 nT_s\right)$
$\exp\left(j\hat{\omega}^K nT_s\right)$
$\hat{x}_{\mathrm{VDLL}}$

Fig. 4. Block diagram of a VDLL receiver.

3.3 Position Domain Joint Tracking Loop (PTL)

The position domain joint tracking loop was introduced in [15] as an extension of the joint tracking loop [10]. The basic idea is to extend the state-vector of the vector DLL by a carrier-phase part. More precisely the PTL state-vector contains for each tracked signal the carrier-phase difference (and its time derivatives) between the carrier-phase of the received signal and the local replica. For each satellite k and carrier-frequency m the vector $\partial\Phi_m^k$ is defined as

$$\partial\Phi_{m,t}^k = \lambda_m \cdot \left(\Delta\varphi_{m,t}^k, \quad \Delta\dot{\varphi}_{m,t}^k, \quad \Delta\ddot{\varphi}_{m,t}^k, \quad \ldots\right)^T,$$
$$\text{with } \Delta\varphi_{m,t}^k = \varphi_{m,t}^k - \hat{\varphi}_{m,t}^k.$$

Accordingly the state-vector used in the PTL loop reads

$$x_{\mathrm{PTL},t} = \begin{pmatrix} \partial\xi_t \\ \partial\Phi_{1,t}^1 \\ \vdots \\ \partial\Phi_{M,t}^1 \\ \partial\Phi_{1,t}^2 \\ \vdots \\ \partial\Phi_{M,t}^K \end{pmatrix}, \; k = 1, \ldots, K, \; m = 1, \ldots, M.$$

Due to the unknown integer ambiguities and bias of the carrier, the carrier-phase and frequency of the local replica is not steered according to $\partial\hat{\xi}_t$ like the spreading code. Instead, the steering is done by introducing a controller that drives the (estimated) carrier-phase differences to zero. And so the final state-space model can be summarized by

$$x_{\mathrm{PTL},t+1} = \Phi_{\mathrm{PTL}} x_{\mathrm{PTL},t} + Bu_t + G_t w_{\mathrm{PTL},t}.$$

The state-transition matrix above, Φ_{PTL} is block diagonal and composed as follows

$$\Phi_{\mathrm{PTL}} = \begin{pmatrix} \Phi_n \otimes I_4 & 0 \\ 0 & I_{(M\cdot K)} \otimes \Phi_n \end{pmatrix},$$

and the process-noise matrix G_t:

$$G_t = \begin{pmatrix} I_4 \otimes I_n \\ \left(H \otimes 1_{M,1} \otimes I_n\right)\pi_4 \end{pmatrix}$$

where $1_{k,l}$ denotes an all-ones matrix of dimension $k \times l$ and π_4 a permutation matrix of form

$$\pi_4 = \begin{pmatrix} 1 & . & . & . & . & . & . & . \\ . & . & . & . & 1 & . & . & . \\ . & 1 & . & . & . & . & . & . \\ . & . & . & . & . & 1 & . & . \\ . & . & 1 & . & . & . & . & . \\ . & . & . & . & . & . & 1 & . \\ . & . & . & 1 & . & . & . & . \\ . & . & . & . & . & . & . & 1 \end{pmatrix}.$$

The control input $u_{\mathrm{PTL},t}$ is the steering input for the carrier NCOs of the different channels. It consists of a phase- and frequency-steering term for every channel. From the derivation of the state-space model it becomes clear that the input-matrix B equals

$$B = \begin{pmatrix} 0_{4,M\cdot K} \\ I_{M\cdot K} \end{pmatrix} \otimes B_n, \text{ with } B_n = \begin{pmatrix} -I_2 \\ 0_{n-2,2} \end{pmatrix},$$

and $0_{k,l}$ denoting an all-zeros matrix of dimension $k \times l$. As the state-vector is not directly available in the receiver, it's estimate is used to determine the carrier-steering input, i.e.

$$u_t = -L\hat{x}_{\mathrm{PTL},t}.$$

Further details on how to determine the control law above can be found e.g. in [21].

Similarly like in the VDLL, the measurement inputs to the Kalman filter are the discriminator outputs of all channels. In the PTL also the carrier-phase discriminator is used. The relationship between the state-vector and the code-phase discriminator is described by Eq. (7). The carrier-phase discriminator output directly relates to the carrier-phase offset and its derivatives for the specific channel, i.e.

$$\begin{aligned} D_{\varphi,m,t}^k &= D_\varphi\left(\partial\Phi_{m,t}^k\right) \\ &\approx \Delta\varphi_{m,t}^k + \frac{T_i}{2}\Delta\dot{\varphi}_{m,t}^k + \frac{T_i^2}{6}\Delta\ddot{\varphi}_{m,t}^k + \ldots + \epsilon_t^k \\ &= H_n \Delta\Phi_{m,t}^k + \epsilon_t^k, \\ \text{with } H_n &= \left(1, \quad \frac{T_i}{2}, \quad \frac{T_i^2}{6}, \quad \ldots, \quad \frac{T_i^{n-1}}{n!}\right). \end{aligned}$$

And so the overall relationship between the state-vector and the measurements reads

$$\begin{aligned} y_{\mathrm{PTL},t} &= \left(\begin{array}{c} D^1_{\tau,1,t} + \hat{\rho}^1_{1,t} \\ \vdots \\ D^1_{\tau,M,t} + \hat{\rho}^1_{M,t} \\ \vdots \\ D^K_{\tau,M,t} + \hat{\rho}^K_{M,t} \\ \hdashline D^1_{\varphi,1,t} \\ \vdots \\ D^K_{\varphi,M,t} \end{array}\right) \\ &= \begin{pmatrix} H_n \otimes H \otimes 1_{M,1} & 0 \\ 0 & I_{M\cdot K} \otimes H_n \end{pmatrix} x_{\mathrm{PTL},t} + v_{\mathrm{PTL},t} \end{aligned} \tag{8}$$

In the above equation, H represents the geometry matrix, defined in the usual manner as consisting of the transposed unit vectors $\vec{e}^{\,k}$ and a column of ones, e.g. [22]. Furthermore, since the state-vector contains the absolute position and clock offset, the estimated pseudoranges $\hat{\rho}^k_{m,t} = c\hat{\tau}^k_{m,t}$ have to be added to the code-phase discriminator outputs (cf. Eq. (6)).

3.4 PTL using the Frequency Discriminator

In the original configuration of the joint tracking receiver, the code- and carrier-phase discriminator outputs are used as measurements [10]. It turns out that this setup is vulnerable in environments where fast and strong fading of the signal happens. The main reason is that if a line-of-sight signal is blocked a strong echo potentially contains a Doppler-shift. Once the line-of-sight signal turns back on, the carrier-phase discriminator observes frequent jumps due to its periodicity. As a consequence the loop possibly steers the frequency of the discussed channel in the wrong direction.

To solve this problem, the additional usage of the frequency discriminator is proposed. The carrier-frequency discriminator has a period large enough and can thus reliably detect a frequency offset which is then corrected by the feedback controller.

Thus the proposed measurement vector reads

$$y_{\mathrm{PTL},t} = \left(\begin{array}{c} D^1_{\tau,1,t} + \hat{\rho}^1_{1,t} \\ \vdots \\ D^K_{\tau,M,t} + \hat{\rho}^K_{M,t} \\ \hdashline D^1_{\varphi,1,t} \\ \vdots \\ D^K_{\varphi,M,t}, \\ \hdashline D^1_{\omega,1,t} \\ \vdots \\ D^K_{\omega,M,t} \end{array}\right).$$

Of course, the measurement covariance matrix has to be updated accordingly to also include the carrier-frequency discriminator's noise variance. Although the carrier-phase and -frequency discriminator use the same correlation results to form the outputs, it can be shown that for small errors in frequency, the covariance between the two is comparably small. It is zero exactly for the case of zero frequency offset. Thus the measurement noise covariance matrix, R_{PTL}, used in the estimator can be modeled as approximately diagonal with the following entries:

$$\begin{aligned} R_\tau &= \frac{2d}{T_i C/N_0} \cdot \left(\frac{c}{f_{\text{code}}}\right)^2, \\ R_\varphi &= \frac{1}{2T_i C/N_0} \cdot \lambda^2, \\ R_\omega &= \frac{2}{\pi^2 T_i^3 C/N_0} \cdot \lambda^2. \end{aligned}$$

Like already described in Eq. (8), the code-phase discriminator is related to the ξ-components of the state-vector. The carrier-phase and now also -frequency discriminator are related to the carrier-phase-components. Finally the overall measurement matrix reads

$$H_{\mathrm{PTL}} = \begin{pmatrix} H_n \otimes H \otimes 1_{M,1} & 0 \\ 0 & I_{M\cdot K} \otimes H_n \\ 0 & I_{M\cdot K} \otimes \begin{pmatrix} 0, & H_{n-1} \end{pmatrix} \end{pmatrix}.$$

It was already pointed out by Van Dierendonck in [23] that the discriminator outputs deviate more and more from the (optimal) linear relationship to the phase or frequency difference at the input with decreasing C/N_0. Additionally, due to the squaring loss the noise of the discriminators increases faster then the C/N_0. Thus at lower signal strength not all discriminators are equally reliable. In the implementation used here the different outputs are only used in specific C/N_0-regions:

$C/N_0 \geq 37$ dB-Hz:	D_τ, D_ω, D_φ
$C/N_0 \in [25, 37)$ dB-Hz:	D_τ, D_ω
$C/N_0 \in [10, 25)$ dB-Hz:	D_τ
$C/N_0 < 10$ dB-Hz:	-

The above modifications can easily be achieved either by removing the corresponding lines in the measurement matrix, or by setting the entries in the measurement covariance matrix to a very large value. Due to numerical stability the first way is preferable.

Finally, the position domain joint tracking receiver is summarized in Fig. 5.

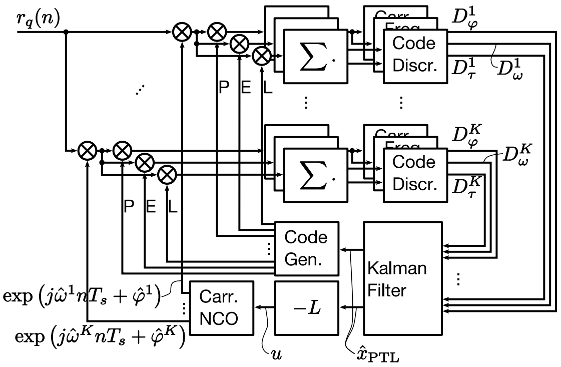


Fig. 5. Block diagram of a PTL receiver.

4 RESULTS

This section shows the results of the three different receiver strategies operating in a pedestrian multipath scenario. After a description of the two considered scenarios the standard receiver architecture is analyzed. It is followed by a comparison with multisatellite algorithms.

4.1 Simulation Environment

Scenario 1 In scenario 1 a pedestrian user was simulated starting with very low velocity and an acceleration up to $2.5\,\mathrm{m/s}$ in totally $60\,\mathrm{seconds}$ (random walk simulation). The skyplot for the first scenario is shown in Fig. 6.

Scenario 2 The second scenario again was simulated with a slowly acceleration pedestrian user (second order polynomial simulation). Seven satellites were available during the $40\,\mathrm{seconds}$, as indicated in the skyplot of Fig. 7.

4.2 Scalar Tracking

Configuration In the standard receiver employing one PLL and (carrier-aided) DLL for each tracked satellite signal, the following configuration was used:

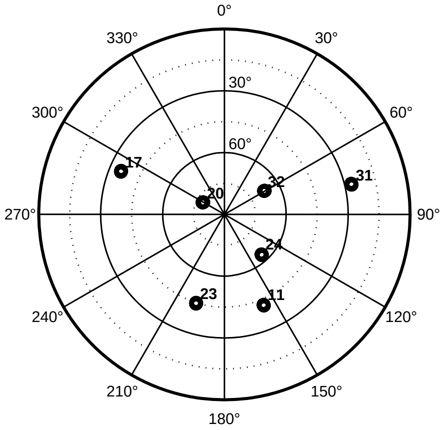


Fig. 6. Skyplot of scenario 1.

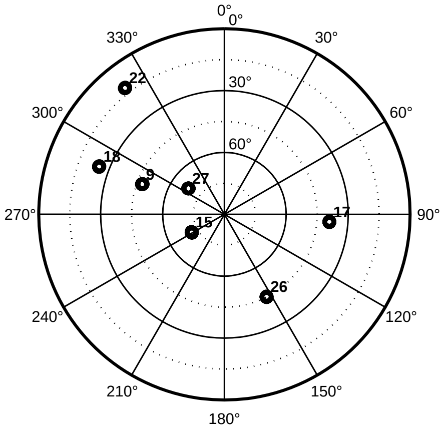
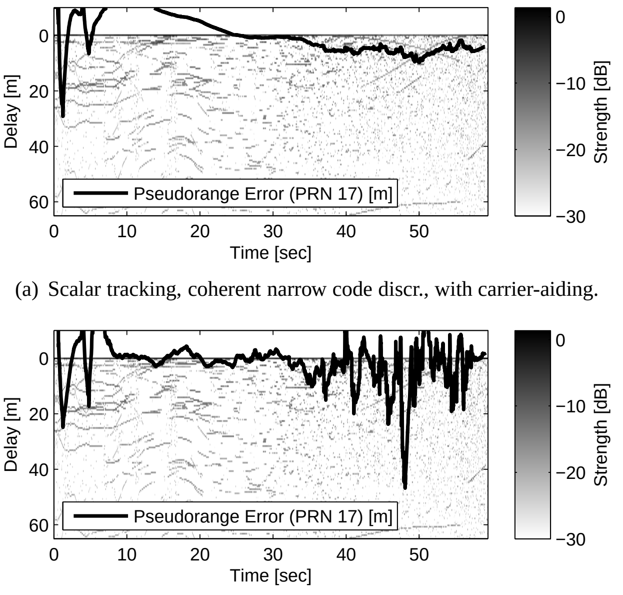


Fig. 7. Skyplot of scenario 2.

Bandwidth DLL (aided):	$B_\tau = 0.1\,\mathrm{Hz}$
Bandwidth DLL (unaided):	$B_\tau = 1\,\mathrm{Hz}$
Noncoherent code-discr.:	Early-late envelope
Coherent code-discr.:	Dot product
Bandwidth PLL:	$B_\varphi = 10\,\mathrm{Hz}$
Carrier-discriminator:	Arctangent
Early-late spacing (wide):	$d = 1\,\mathrm{Chip}$
Early-late spacing (narrow):	$d = 0.25\,\mathrm{Chips}$

The minimum early-late spacing was chosen according to $d = 1/\beta$, with β the (two-sided) frontend bandwidth in Megahertz [20]. In the current setup a small bandwidth of $\beta = 4$ was simulated, as this is seen typical for low-cost commercial receivers.

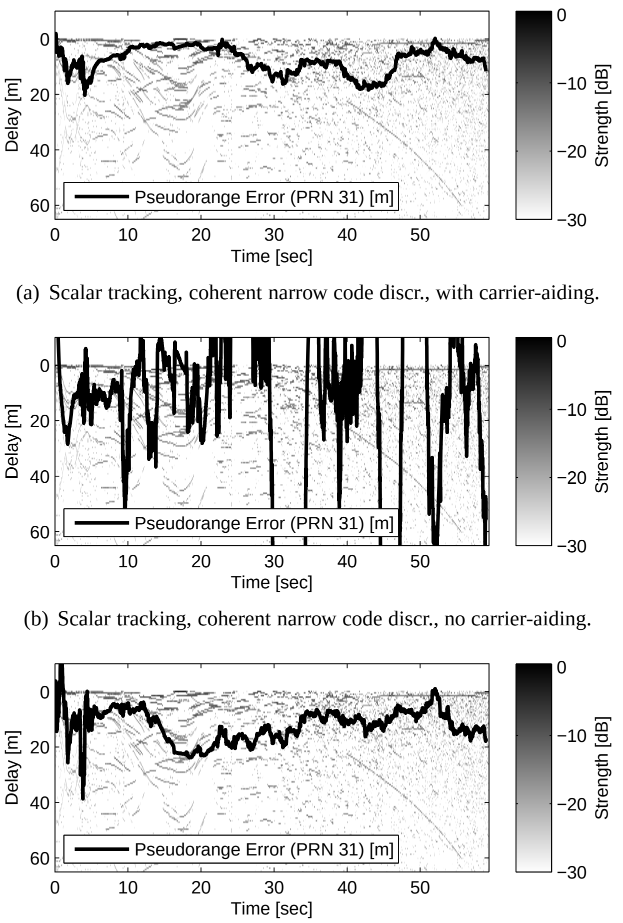
Tracking Fig. 8 illustrates the tracking behavior of the scalar tracking module with and without carrier-aiding for PRN 17. Of course, the tracking error of the carrier-aided loops – Fig. 8(a) – is smoother than the one without aiding – Fig. 8(b). On the other hand, the aided loops take longer to settle once an offset due to long existing reflections oc-



(a) Scalar tracking, coherent narrow code discr., with carrier-aiding.

(b) Scalar tracking, coherent narrow code discr., no carrier-aiding.

Fig. 8. Scenario 1: Pseudorange error analysis for PRN 17 in relation to the channel impulse response (delay and strength of the echoes), $T_i = 10\,\mathrm{ms}$.



(a) Scalar tracking, coherent narrow code discr., with carrier-aiding.

(b) Scalar tracking, coherent narrow code discr., no carrier-aiding.

(c) Scalar tracking, coherent wide code discr., with carrier-aiding.

Fig. 9. Scenario 1: Pseudorange error analysis for PRN 31 tracked with a scalar DLL/PLL in relation to the channel impulse response (delay and strength of the echoes), $T_i = 10\,\mathrm{ms}$.

cured. A problem that the scalar tracking loops face can be seen around $50\,\mathrm{s}$: Although the line-of-sight signal is comparably strong, the DLL output could be biased because of the echoes, especially echoes with medium delay.

The evaluation of the pseudorange tracking for PRN 31 in Fig. 9 demonstrates the difficulty the scalar tracking loops face, when the line-of-sight signals are strongly attenuated or completely blocked. Without carrier-aiding the pseudorange error is likely to grow up to tens of meter, see Fig. 9(b). On the other hand, the carrier-aided loop remains closer to the actual code-phase, as illustrated in Figs. 9(a) and 9(c). Additionally, comparing Figs. 9(a) and 9(c), it can be seen that a small correlator spacing may improve the code-phase tracking performance if some echoes exist.

On the other hand, a drawback of the narrow correlator spacing is shown in Fig. 10 (PRN 11). Although, as previously seen for PRN 31, the code-tracking precision might be enhanced using a narrow early-late spacing (Fig. 9), it loses robustness compared to a wider spacing. Therefore in a narrow configuration the DLL is more likely losing lock.

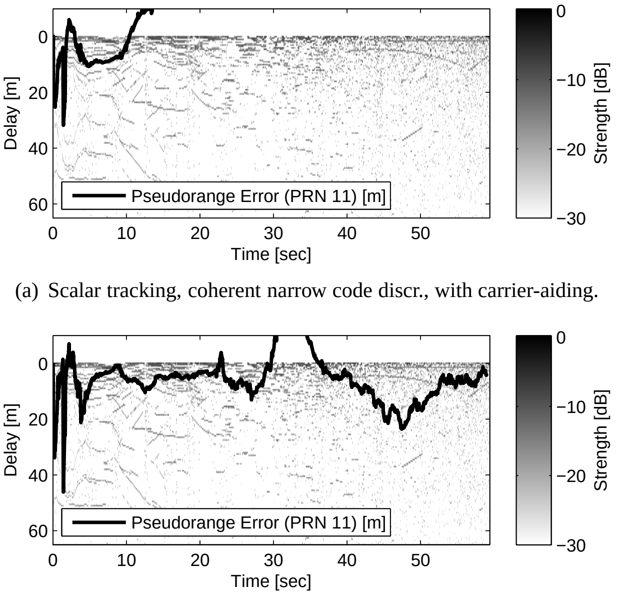
Positioning Finally, Fig. 11 shows the absolute error in the position estimate for two scalar tracking receiver. One used a narrow correlator spacing, the other a wider spacing. The state-space model used in the Kalman filter to estimate the position was the same like the one used by the multi-satellite receivers. It consisted of three independent second order processes for the receiver movements in ECEF x-, y- and z-direction and a third order process for the receiver clock bias. The corresponding modeled strengths of the process noise were

Receiver movements:	$\sigma_{\dot{r}}^2 = 600\,\mathrm{m}^2/\mathrm{s}^3 \cdot T_i$
Receiver clock bias:	$\sigma_{c\cdot\ddot{\delta t}}^2 = 150\,\mathrm{m}^2/\mathrm{s}^5 \cdot T_i$

Both receivers were configured with carrier-aiding. Whenever a DLL temporarily lost lock, its measurement, i.e. pseudorange, was excluded from the position estimating Kalman filter to prevent divergence.

4.3 Vector Delay Locked Loop

Configuration The state-space model used in the VDLL is the same as also used by the scalar-tracking receivers in



(a) Scalar tracking, coherent narrow code discr., with carrier-aiding.

(b) Scalar tracking, coherent wide code discr., with carrier-aiding.

Fig. 10. Scenario 1: Pseudorange error analysis for PRN 11 tracked with a scalar DLL/PLL in relation to the channel impulse response (delay and strength of the echoes), $T_i = 10\,\mathrm{ms}$.

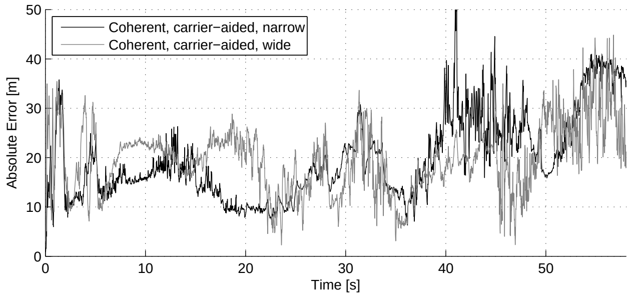


Fig. 11. Scenario 1: Positioning error of two scalar-tracking receivers, applying carrier-aiding of the DLL and Kalman filtering for the position estimation ($T_i = 10\,\mathrm{ms}$).

the previous paragraph and not repeated here. The configuration of the code-phase discriminator was as follows

Noncoherent code-discr.:	Early-late envelope
Early-late spacing (wide):	$d = 1\,\mathrm{Chip}$
Early-late spacing (narrow):	$d = 0.25\,\mathrm{Chips}$

In all multisatellite algorithms a crucial step is the weighting of the measurements, i.e. the discriminator outputs. As already outlined above, the weighting is achieved by measuring the carrier-to-noise density ratio C/N_0 for the tracked signals. Since the VDLL is not aiming at a carrier-phase lock, the algorithm of Benedict and Soong [24] was applied in the C/N_0-meter. It only needs the squared magnitude of the correlation results for the determination of C/N_0 and is thus less sensitive to small carrier-frequency errors, as compared to the algorithm proposed by Van Dierendonck in [23].

Tracking For the same satellites and scenario as already discussed in the previous section 4.2, a comparison between the true pseudorange and its estimate is shown in Fig. 12. It has to be kept in mind that the VDLL doesn't actually track the pseudorange. But given the estimated receiver position and clock offset, the pseudorange is estimated for the replica generation. This value is illustrated in Fig. 12. The error is less smooth and could grow larger than in the carrier-aided DLL. But an analysis of the different channels shows that the lock is more robust in a way that none of the satellites was lost in the analysis.

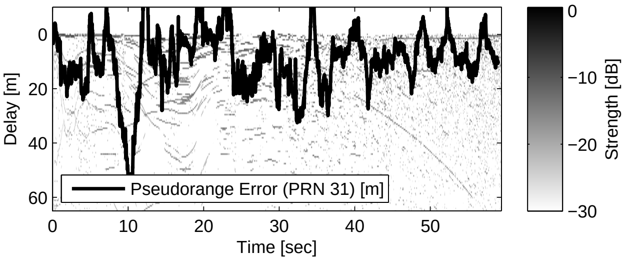


Fig. 12. Scenario 1: Pseudorange error analysis for PRN 31 tracked by a VDLL in relation to the channel impulse response (delay and strength of the echoes), $T_i = 10\,\mathrm{ms}$.

Positioning Fig. 13 compares the positioning error of two VDLL receivers. The first one used a narrow the second a wide early-late correlator spacing. The first receiver with narrow spacing loses lock already after a few seconds of processing, whereas the second receiver is more robust. But the positioning performance is comparable to the one of the scalar tracking receivers.

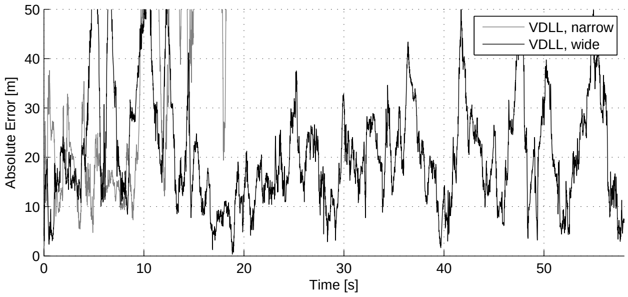


Fig. 13. Scenario 1: Positioning error of two VDLL receivers ($T_i = 10\,\mathrm{ms}$).

4.4 Joint Tracking Loop

Configuration The configuration of the PTL closely followed the one of the VDLL as already outlined in the previous section 4.3. Additionally the following carrier-phase and carrier-frequency discriminators were used:

Coherent code-discr.:	Dot product
Carrier-frequency discr.:	Four-quadrant arctangent
Carrier-phase discr.:	Arctangent

Tracking The pseudorange analysis is also performed for the PTL receiver. In Fig. 14 it's shown that the tracking error is smoothened, compared to the VDLL. Furthermore the error is smaller than in the case of the VDLL and less biased than in a scalar-tracking receiver.

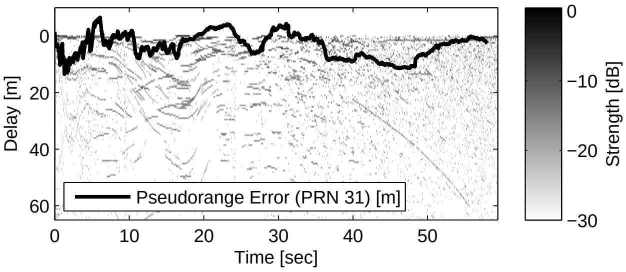


Fig. 14. Scenario 1: Pseudorange error analysis for PRN 31 tracked by a PTL in relation to the channel impulse response (delay and strength of the echoes), $T_i = 10\,\text{ms}$.

Positioning The positioning error of the PTL loop is decreased as compared to the VDLL and the scalar-tracking receiver (Fig. 15). It turns out that the PTL positioning is affected by the multipath propagation, but less than the other algorithms. Or at least not more than other algorithms.

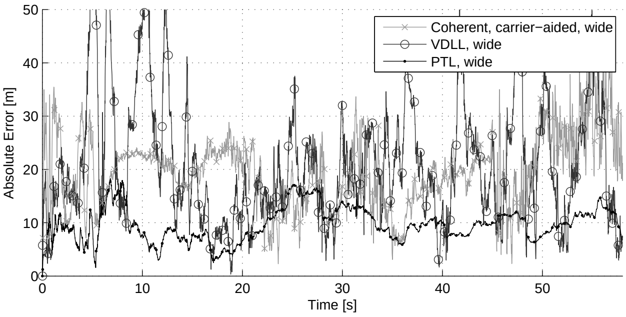


Fig. 15. Scenario 1: Positioning error of three different receivers: Scalar tracking, VDLL and PTL ($T_i = 10\,\text{ms}$).

The same receiver algorithms were tested with the second multipath scenario (*scenario 2*). The positioning error of the algorithms is shown in Fig. 16. Like already observed in scenario 1, the PTL position estimate doesn't vary as much as with the other two considered receiver types. But still it can adapt itself fast to a changing environment, like seen in the figure around 5–15 seconds. In this interval all satellites are received at high signal strength and with only small echo contributions. Therefore the VDLL and PTL receiver show small errors, whereas carrier-aided loops need more time to settle to the true value.

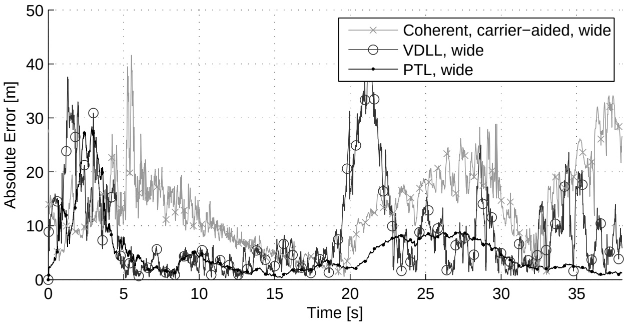


Fig. 16. Scenario 2: Positioning error of three different receivers: Scalar tracking, VDLL and PTL ($T_i = 10\,\text{ms}$).

5 DISCUSSION

Although proposed mainly for reducing the multipath error [4], a narrow correlator spacing does not always result in a better performance compared to a wider spacing. It's shown that it potentially reduces the multipath bias, but on the other, reduces the robustness of the receiver. The results indicate that a receiver aiming at high accuracy code-phase tracking and high robustness achieves the best results if a wider early-late spacing is chosen and the DLL is setup with carrier-aiding. This setup shows to be successful especially in situations of fast-fading. Where in the situation of long existing echoes, this setup suffers from the very small bandwidth of the DLL and is thus vulnerable to a multipath bias.

As already outlined in [13], the vector delay locked loop has an enhanced robustness compared to a receiver employing scalar tracking loops. But this is not true, as seen in Fig. 13, if the early-late correlator spacing is chosen too small. In this case the loop might still lose lock. This can be interpreted as follows: The modeling of the measurements assumed that the relationship between code-phase error and the discriminator output is linear. But once the error is outside of the interval $[-dT_c/2, dT_c/2]$ linearity is lost. Consequently the loop is not able to steer enough. Especially in situations where few line-of-sight signals are blocked or strongly attenuated, this could then cause to VDLL to lose lock. On the other hand, choosing d to be small could enhance the accuracy of the receiver when the signals are all received with high strength.

The benefit of including the carrier-phases in multisatellite tracking loops was already demonstrated for an open-sky scenario [15]. The results in the previous sections show that the PTL receiver achieves the same robustness as the VDLL receiver. Additionally, the carrier-phase discriminator error due to multipath propagation is limited to approximately 4.8 cm [25], whereas the error of the code could grow to tens of meters. Therefore the PTL is less sensitive to short-term existing echoes than a VDLL. The same benefit of the carrier-phase is also seen in the carrier-aided code-phase tracking.

A big drawback of the inclusion of the carrier in the tracking is the long settling time. This has a negative impact whenever the environment is approximately static and multipath propagation occurs. If also the loop estimate diverges from the true value (either code-phase or position directly), the carrier-aided loops take longer to reach a zero error again. Two examples can be seen in the analyses: For a scalar tracking receiver in Fig. 8(a) between 10 and 20 $\mathrm{seconds}$, and for a multisatellite receiver in Fig. 16 between 25 and 30 $\mathrm{seconds}$.

The constraint of keeping the computational complexity as low as possible has been achieved by only using the typical three (complex) correlators per channel: early, late and prompt. This is what already exists in low-cost receivers. Among the three different considered algorithms, the PTL has the highest computational demands. In all of the three a Kalman filter is applied for the position estimation. The scalar tracking receiver and the VDLL use the same state-space model with a complexity of $\mathcal{O}\left(n^3+K^3\right)$, where n denotes the highest model order among the position and clock bias processes and K the number of tracked satellites. The PTL has an increased state-space dimension and thus a higher complexity with $\mathcal{O}\left(n^3K^3\right)$. Additionally all the multisatellite algorithms have the difficulty of requiring a synchronization of the correlators due to the loop closure via the Kalman filter.

6 CONCLUSION

The figures in this paper show that multisatellite algorithms provide a high robustness also in multipath environments. But a crucial step is the choice of receiver configuration parameters, like the early-late correlator spacing d and the weighting of the measurements. Furthermore in coherent multisatellite algorithms like the joint tracking loop it is important to reject measurements if the signal strength of a channel drops below a certain threshold. Although having a high robustness, the VDLL can only achieve a comparable or even worse positioning performance compared to a standard receiver using carrier-aided code-phase tracking. The best average positioning performance is obtained using a coherent multisatellite algorithm like the PTL.

ACKNOWLEDGEMENTS

The authors would like to thank the German Federal Ministry of Economic Affairs and Technology (BMWi) and the German Aerospace Center (DLR) for a financial grant (FKZ: 50NA0911).

REFERENCES

[1] G. W. Hein, J.-C. Martin, J.-L. Issler, J. Godet, P. Erhard, R. Lucas-Rodriguez, and T. Pratt, “Galileo frequency and signal design,” *GPS World*, pp. 30–37, June 2003.

[2] C. C. Counselman, III, “Multipath-rejecting GPS antennas,” *Proceedings of the IEEE*, vol. 87, pp. 86–91, Jan. 1999.

[3] M. Cuntz, M. Heckler, S. Erker, A. Konovaltsev, M. Sgammini, A. Hornbostel, A. Dreher, and M. Meurer, “Navigating in the Galileo test environment with the first GPS/Galileo multi-antenna-receiver,” in *Proc. 5th ESA Workshop on Satellite Navigation Technologies (NAVITEC)*, Dec. 2010.

[4] A. J. Van Dierendonck, P. Fenton, and T. Ford, “Theory and performance of narrow correlator spacing in a GPS receiver,” *NAVIGATION, Journal of the Institut of Navigation (ION)*, vol. 39, pp. 265–283, Fall 1992.

[5] V. A. Veitsel, A. V. Zhdanov, and M. I. Zhodzishsky, “The mitigation of multipath errors by strobe correlators in GPS/GLONASS receivers,” *GPS Solutions*, vol. 2, pp. 38–45, 1998.

[6] M. Irsigler and B. Eissfeller, “Comparison of multipath mitigation techniques with consideration of future signal structures,” in *Proc. ION GPS/GNSS*, pp. 2584–2592, Sept. 2003.

[7] J. Soubielle, I. Fijalkow, P. Duvaut, and A. Bibaut, “GPS positioning in a multipath environment,” *IEEE Transactions on Signal Processing*, vol. 50, pp. 141–150, Jan. 2002.

[8] R. D. van Nee, J. Siereveld, P. C. Fenton, and B. R. Townsend, “The multipath estimating delay lock loop: approaching theoretical accuracy limits,” in *Proc. IEEE Position Location and Navigation Symposium*, pp. 246–251, Apr. 1994.

[9] B. Krach, P. Robertson, and R. Weigel, “An efficient two-fold marginalized bayesian filter for multipath estimation in satellite navigation receivers,”

EURASIP Journal on Advances in Signal Processing, vol. 2010, Feb. 2010.

[10] K. Giger, P. Henkel, and C. Günther, “Joint satellite code and carrier tracking,” in *Proc. ION Internationl Technical Meeting ITM*, pp. 636–645, Jan. 2010.

[11] E. M. Copps, G. J. Geier, W. C. Fidler, and P. A. Grundy, “Optimal processing of GPS signals,” *NAVIGATION, Journal of the Institut of Navigation (ION)*, vol. 27, pp. 171–182, Fall 1980.

[12] J. J. Spilker Jr., “Fundamentals of signal tracking theory,” in *Global Positioning System: Theory and Applications* (B. W. Parkinson and J. J. Spilker Jr., eds.), vol. 1, pp. 245–327, AIAA, Inc., 1996.

[13] M. Lashley, D. M. Bevly, and J. Y. Hung, “A valid comparison of vector and scalar tracking loops,” in *Proc. IEEE/ION Position, Location and Navigation Symposium (PLANS)*, pp. 464–474, May 2010.

[14] K. Giger, P. Henkel, and C. Günther, “Multifrequency multisatellite carrier tracking,” in *Proc. Fourth European Workshop on GNSS Signals and Signal Processing*, Dec. 2009.

[15] K. Giger and C. Günther, “Position domain joint tracking,” in *Proc. 5th ESA Workshop on Satellite Navigation Technologies (NAVITEC)*, Dec. 2010.

[16] A. Lehner and A. Steingass, “A novel channel model for land mobile satellite navigation,” in *Proc. ION GNSS 2005*, pp. 2132–2138, Institute of Navigation (ION), Sept. 2005.

[17] C. J. Hegarty, “Analytical model for GNSS receiver implementation losses,” *NAVIGATION, Journal of the Institut of Navigation (ION)*, vol. 58, no. 1, pp. 29–44, 2011.

[18] R. M. Gray, “Quantization noise spectra,” *IEEE Transactions on Information Theory*, vol. 36, pp. 1220–1244, Nov. 1990.

[19] P. W. Ward, J. W. Betz, and C. J. Hegarty, “Satellite signal acquisition, tracking, and data demodulation,” in *Understanding GPS, Principles and Applications* (E. D. Kaplan and C. J. Hegarty, eds.), pp. 153–241, Artech House, Inc., 2006.

[20] J. W. Betz and K. R. Kolodziejski, “Extended theory of early-late code tracking for a bandlimited GPS receiver,” *NAVIGATION, Journal of the Institut of Navigation (ION)*, vol. 47, no. 3, pp. 211–226, 2000.

[21] R. F. Stengel, *Optimal Control and Estimation*. Dover Publications, Inc., 1994.

[22] G. Strang and K. Borre, *Linear Algebra, Geodesy, and GPS*. Wellesley-Cambridge Press, 1997.

[23] A. J. Van Dierendonck, “GPS receivers,” in *Global Positioning System: Theory and Applications* (B. W. Parkinson and J. J. Spilker Jr., eds.), vol. 1, pp. 329–407, AIAA, Inc., 1996.

[24] T. R. Benedict and T. T. Soong, “The joint estimation of signal and noise from the sum envelope,” *IEEE Transactions on Information Theory*, vol. 13, pp. 447–454, July 1967.

[25] M. S. Braasch, “Multipath effects,” in *Global Positioning System: Theory and Applications* (B. W. Parkinson and J. J. Spilker Jr., eds.), vol. 1, pp. 547–568, AIAA, Inc., 1996.