Performance of a Partially Coherent Ultra-Tightly Coupled GNSS/INS Pedestrian Navigation System Enabling Coherent Integration Times of Several Seconds to Track GNSS Signals Down to 1.5 dBHz.

Thomas Pany[(1)], Jón Winkel[(1)], Bernhard Riedl[(1)], Markus Restle[(1)], Thomas Wörz[(3)], Robert Schweikert[(3)], Herbert Niedermeier[(2)], Gerald Ameres[(2)], Bernd Eissfeller[(2)], Stefano Lagrasta[(4)] and Gustavo López-Risueño[(5)]

[(1)]IFEN GmbH, Alte Gruber Straße 6, 85586 Poing, Germany,
E-mail: T.Pany@ifen.com
[(2)] University FAF Munich, Institute of Geodesy and Navigation, 85577 Neubiberg Germany
Email: Herbert.Niedermeier@unibw.de
[(3)] AUDENS ACT Consulting GmbH, Argelsrieder Feld 22, 82234 Wessling, Germany
Email: Thomas.Woerz@audens.com
[(4)] Telespazio S.p.A., via Tiburtina, 965, 00156 Roma, Italy
Email: stefano.lagrasta@telespazio.com
[(5)] ESA/ESTEC, Keplerlaan 1, 2200 AG Noordwijk, The Netherlands
Email: Gustavo.Lopez.Risueno@esa.int

BIOGRAPHIES

Dr. Thomas Pany works for IFEN GmbH as a senior research engineer in the GNSS receiver department. In particular, he is concerned with algorithm development and C/C++/assembler coding. He was for six years assistant professor (C1) at the University FAF Munich and for four years research associate at the Space Research Institute of the Austrian Academy of Science. His research interests include GNSS receivers, GNSS/INS integration, signal processing and GNSS science.

Dr. Jón Winkel is head of receiver technology at IFEN GmbH since 2001. He studied physics at the universities in Hamburg and Regensburg. He received a PhD (Dr.-Ing.) from the University of the Federal Armed Forces in Munich in 2003 on GNSS modeling and simulations.

Bernhard Riedl received his diploma in Electrical Engineering and Information Technology from the Technical University of Munich. Since 1994 he has been concerned with research in the field of real-time GNSS applications at the University FAF Munich. In 2006 he joined IFEN GmbH, where he is currently working as product development manager.

Markus Restle is a Systems Engineer at IFEN GmbH focusing on software development for Embedded Systems. He holds a Dipl-Ing(FH) in Communications Engineering from the University of Applied Sciences in Konstanz, Germany.

Dr. Thomas Wörz works for AUDENS ACT GmbH as managing director and senior consultant. He worked for 10 years at the German Aerospace Centre (DLR) as research associate, before he co-founded AUDENS ACT in 1998. His expertise comprises satellite system design and corresponding signal processing for communication and navigation systems, such as analysis and modeling of (mobile) satellite channels, signal structures and corresponding receiver design, including channel estimation, equalization, synchronization, demodulation, decoding and network aspects.

Dr. Robert Schweikert works for AUDENS ACT GmbH as managing director and senior consultant. Before he co-founded AUDENS ACT in 1998, he was 20 years with the German Aerospace centre and headed the 'Communication Theory' department for more than 5 years after a period of more than 14 years in which he had been head of the 'Communication Systems' department. In addition he has also been deputy head of the Institute. His expertise comprises satellite system design and corresponding signal processing for communication and navigation systems, such as analysis and modeling of (mobile) satellite channels, signal structures and corresponding receiver design, including channel estimation, equalization, synchronization, demodulation, decoding and network aspects.

Herbert Niedermeier is a mechanical engineer in the field of aeronautical engineering and research associate at the Institute of Geodesy and Navigation at the University FAF Munich. He is working on sensor fusion for different applications, e.g. pedestrian or vehicle navigation, with emphasis on airborne gravimetry. Current research is focused on inertial navigation and sensor integration, particularly for pedestrian and vehicle navigation.

Gerald Ameres was working at the Institute of Geodesy and Navigation for seven years. He has done simulation, evaluation and performance testing of various IMU systems and its integration with other navigation sensors, mainly GNSS. In the field of GNSS he has made contributions to the real-time pre-processing of GNSS measurements for integrity evaluations.

Prof. Bernd Eissfeller is Full Professor and Director of the Institute of Geodesy and Navigation at the University FAF Munich. He is responsible for teaching and research in the field of navigation and signal processing. Till the end of 1993 he worked in industry as a project manager on the development of GPS/INS navigation systems. From 1994 - 2000 he was head of the GNSS Laboratory and since 2000 full professor of Navigation at University FAF. He is author of more than 215 scientific and technical papers.

Stefano Lagrasta has a Master Degree in Electronics Engineering and leads the Navigation Systems and Services engineering unit at Telespazio, with former experience on design and implementation of satellite Attitude Control and ground Flight Dynamics Systems.

Dr. Gustavo López-Risueño has a PhD (2003) and a M Sc. (1998) in Telecommunications Engineering, from the Universidad Politécnica de Madrid, Spain. He has held several positions in academia, ESA and NC3A (NATO). Currently, he is a Radionavigation Systems Engineer at ESA/ESTEC working mainly on signal processing for GNSS receivers and monitoring.

ABSTRACT

The paper describes a prototype of an ultra-tightly coupled GNSS/INS system. A MEMS IMU is used and it is argued that the canonical way of ultra-tight coupling (i.e. an IMU strapdown navigation solution plus an – IMU bias estimating – error state Kalman filter, which is updated with GNSS correlation values) is difficult to realize due to the MEMS gyro bias instability. Instead the strapdown calculation is replaced by a step detection and dead reckoning algorithm and the error state Kalman filter is updated with (non-coherent) code pseudoranges. To aid signal processing, the strapdown calculation is maintained but the strapdown trajectory is sub-divided into segments of e.g. 100 ms – 2 s duration. Those segments aid signal processing in a coherent way and allow (together with assistance data and a stable oscillator) coherent integration times of the same duration. We will argue that the long coherent integration mitigates three major problems of indoor signal processing: squaring loss, multipath and cross-correlations and allows tracking GNSS signals of around 1.5 dBHz or below. The paper also contains a technical description of the built prototype and presents end-to-end performance results for simulated Galileo E1 signals and real GPS C/A signals.

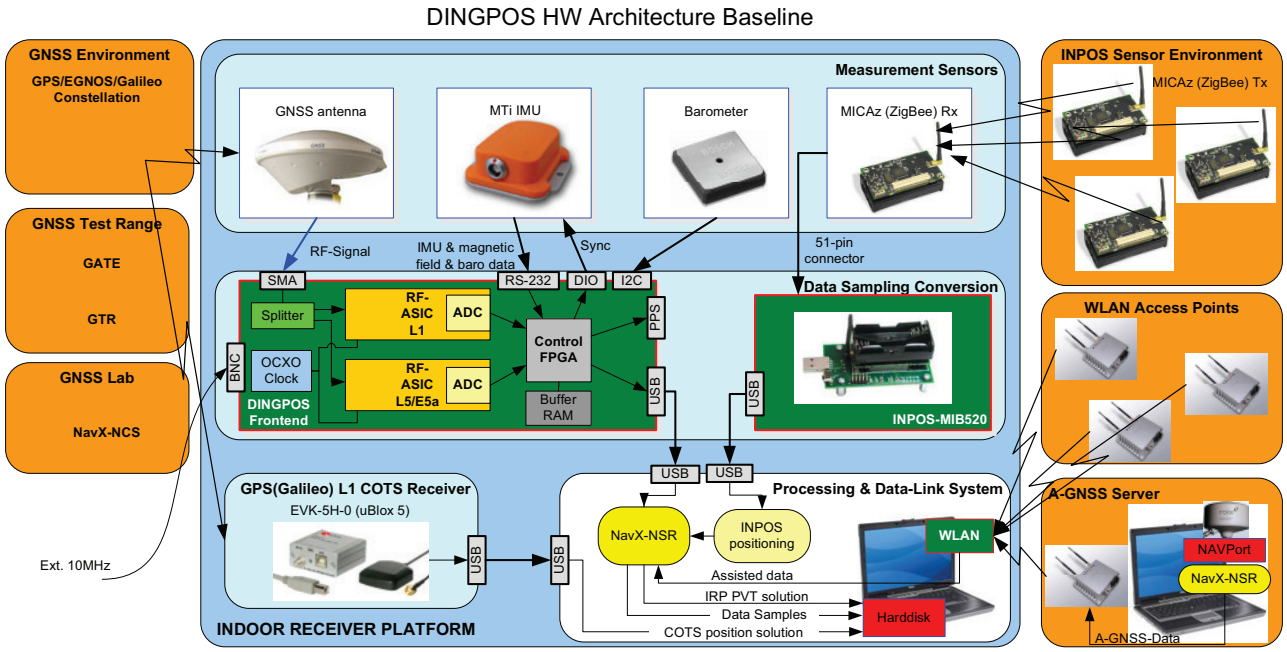

Fig. 1 *Block diagram of the DINGPOS system*

INTRODUCTION

ESA funded a newly developed GNSS/INS navigation system to investigate i) the benefit of Galileo signals and current GPS signals under deployment for indoor positioning, and ii) to investigate new indoor positioning methods for pedestrians based on those signals. It may have applications as a pedestrian navigation system (PNS) for emergency forces and in the military domain. The system is under development since 2007 and a prototype is running since 2008. At the moment, the system is optimized and being prepared for the finals test in the German Galileo Test Bed (GATE). One of the key features of this system is the support of a *coherent integration* over several seconds.

The system uses as sensors a L1/L5 GNSS software receiver, a MEMS IMU including a magnetometer and a barometer, WiFi (=WLAN) power readings as well as a ZigBee based radio navigation system. A block diagram is shown in Fig. 1 and a list of the major components is contained in Tab. 1. The NavX-NSR V2.0 software receiver acts as integration platform. It decodes the GNSS signals at E1=L1 and E5a=L5 as well as the IMU, magnetometer, barometer, WiFi and ZigBee data. The processing of the data can be done in real-time or in post-processing. The software receiver itself was configured to process the signals listed in Tab. 2. The algorithms used for the DINGPOS processing were implemented in C or C++ and were loaded into the NavX-NSR as dynamic link libraries (DLLs). A second software receiver acts as reference station to provide assistance data (incl. data bits), coarse start position and time synchronization via the network time protocol (NTP).

The roving software receiver is called indoor receiver platform (IRP) and is integrated onto a backpack (cf. Fig. 2).

Tab. 1 *DINGPOS components*

	Name	Company
Software receiver	NavX-NSR V2.0	IFEN GmbH
IMU with magnetometer	MTi	Xsens
GNSS frontend	NAVport 2 with integrated barometer and OCXO	IFEN GmbH
Antenna	Zephyr 2	Trimble
WLAN	Integrated with Laptop	Dell
ZigBee	INPOS	Telespazio

The following previous publications describe other aspects of the DINGPOS system: the PNS and the constrained strapdown is covered in [1,2], signal processing is part of [3]. Preliminary results of a parallel study can be found in [4].

Tab. 2 *Processed GNSS Signals*

	3 dB signal bandwidth	Signal components
GPS C/A	4 MHz	Data
Galileo/GATE E1	4 MHz	Data+Pilot
GPS L5	16 MHz	Data+Pilot
Galileo/GATE E5a	16 MHz	Data+Pilot

The remainder of the paper is as follows: First, the GNSS/INS integration algorithms is described and contrasted to the canonical approach. Then, the key-components necessary to build the prototype are outlined and the signal processing benefits are discussed. Galileo E1 performance results are presented for connecting the IRP to the RF signal simulator NavX-NCS and for a real run through the IFEN office building while receiving GPS C/A code signals.

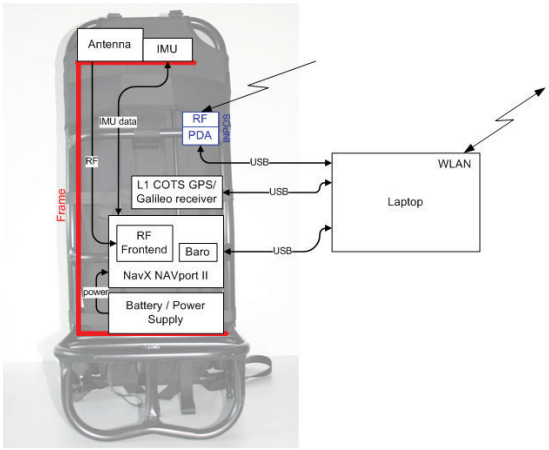

Fig. 2 *Mounting of the IRP*

GNSS/INS INTEGRATION ALGORITHM

Ultra-tight coupling (or deep coupling) of GNSS and IMU data is a possible integration method and is covered in many research publications and patents [5,6]. The core elements of this method are

- a strapdown calculation for IMU data processing
- and an error state Kalman filter using GNSS observations.

The strapdown algorithm computes a user trajectory (after a coarse/fine alignment procedure) and the Kalman filter estimates the error of this trajectory with respect to the true trajectory. The Kalman filter fully controls the GNSS correlation process by providing code and carrier numerically controlled oscillator (NCO) rate and phase values. By linking all GNSS channels via the Kalman filter, *vector* tracking is realized. Vector tracking may come in several variants depending on the used observations and Kalman state vector [7].

Coherent GNSS/INS Integration

Ultra-tight coupling can be done in a fully coherent way as shown in Fig. 3 and as discussed in other works like [8]. Fig. 3 shows a simplified data flow diagram and emphasizes *coherent data* as red lines. We call all data coherent that is dependent on the GNSS carrier phase or influences the GNSS carrier phase. On the signal processing side, this includes the IF samples, (I/Q) correlation values and the estimated carrier phase. The example of Fig. 3 uses correlation values to update the Kalman filter.

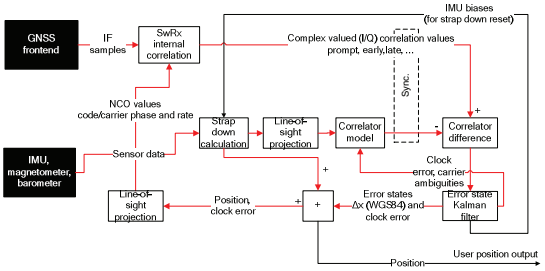

Fig. 3 *Fully coherent ultra-tight GNSS/INS integration (red lines denote coherent data)*

To maintain coherency in the whole system, the Kalman filter must be able to predict the phase of the complex valued GNSS correlation values.

The prediction needs to be carried out *over the considered GNSS correlation time*. During this time no measurements are available and the Kalman system model needs to be *accurate* over this period. Only then, the observed-minus-modeled correlator difference can be approximated by a linear observation equation. If the prediction is inaccurate and linearization is not possible, we need to estimate the phase from the measurements itself, which is not possible for low SNR values; the system losses its coherency.

This prediction not only involves prediction of residual user position and clock errors, but also prediction of atmospheric and hardware delays. The latter ones might be derived from a carrier phase offset state in the Kalman filter that also models the carrier phase ambiguity. This fully coherent setup computes modeled GNSS correlation values and subtracts those values from the real correlation values. The difference is used to update the Kalman filter. Provided that the difference is small, it can be shown that this integration scheme is *optimal* in terms of unbiasedness and minimum variance of the estimated parameters. The correlator difference depends on the predicted-minus-observed carrier phase difference, which should be less than a fraction of the GNSS carrier wave length (i.e. maximally a few cm). The Kalman filter also estimates IMU parameters (e.g. gyro and accelerometer biases) that can be fed into the strapdown calculation.

Non-Coherent GNSS/INS Integration

It is a well known fact that a MEMS IMU gives a strapdown trajectory with large errors due to gyro bias variations. This makes it *impossible* to predict the residual user position error with a (relative) accuracy at the centimeter level over the considered coherent integration time of 2 s; the concept of Fig. 3 cannot be used with a MEMS IMU.

On the other hand, the work [1,2] nicely shows that the Xsens IMU plus magnetometer data can be used to detect steps of a walking human and to estimated stride and heading of those steps. With a start position, a dead reckoning system can be setup, providing a much more stable user trajectory compared to a (MEMS) strapdown calculation. The dead reckoning trajectory has a relative accuracy at the decimeter level, but is not related to the GNSS carrier phase (because the user steps do *not* have a fixed geometric relationship to the antenna motion). Typical errors of the dead reckoning trajectory – like start position errors, or a heading bias – can be well estimated by an error state Kalman filter using GNSS code pseudoranges. A possible ultra-tight integration scheme of a dead-reckoning solution with GNSS code pseudoranges is depicted in Fig. 4. This scheme is *non-coherent* because the pseudoranges and the NCO feedback values are not related to the carrier phase. Most importantly, the Kalman filter does *not* control the phase of the carrier NCO, only the rate (=Doppler) is set by the Kalman filter. This limits the coherent integration time to be less than 100 ms (otherwise the non-linear user motion would corrupt the correlation result, see below or cf. [9]).

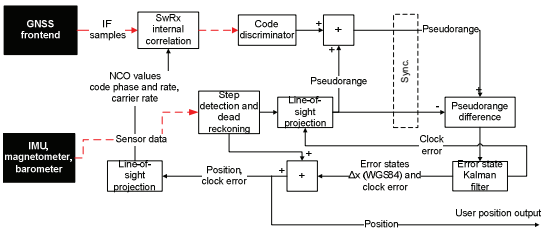

Fig. 4 *Non-coherent ultra-tight GNSS/INS integration (red lines denote coherent data, dashed red lines denote short term coherent data)*

Partially Coherent GNSS/INS Integration

The key idea of the newly proposed integration scheme is to combine a strapdown calculation and a step detection into a new trajectory type (called *μ-trajectory*). The μ-trajectory is composed of piece-wise smooth segments. Start- and end-points of the segments are obtained from interpolating the dead reckoning solution (cf. Fig. 5). The segments are aligned with the coherent integration intervals of signal processing.

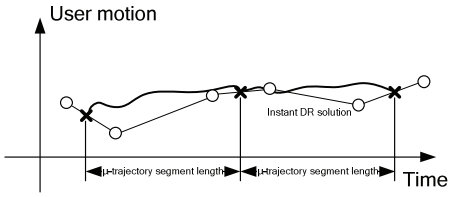

Fig. 5 *µ-trajectory fitted into instant DR solution (circles denote detected steps, crosses are interpolated positions)*

For each segment the mean strapdown acceleration and velocity is adjusted in order that the strapdown trajectory connects the start and end-point.

We will demonstrate that by constraining the strapdown calculation in this way, we can estimate the *non-linear* IMU (= antenna) motion with centimeter accuracy. In other words, the µ-trajectory can be used to predict the carrier phase of single segments. The resulting ultra-tight integration scheme is depicted in Fig. 6. It emphasizes the fact that the phase of the carrier NCO is now also controlled by the Kalman filter during the µ-trajectory segments. Together with a data-wipe off and a stable oscillator, this algorithms allows using of coherent integration times up to 2 s.

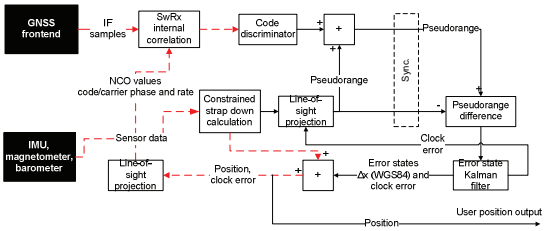

Fig. 6 *Mixed ultra-tight GNSS/INS integration with non-coherent measurement update and coherent feedback (red lines denote coherent data, dashed red lines denote short term coherent data)*

Implementation of a Partially Coherent Integration

The realization of an ultra-tight GNSS/INS system is difficult, because it requires synchronizing of different data before processing. The processing itself introduces latencies and suitable data buffers are required to accommodate them.

The block diagram of the DINGPOS scheme is shown in Fig. 7. It extends Fig. 6 and includes other positioning methods based on WiFi (=WLAN) or ZigBee signal power measurements. Furthermore, the dead reckoning solution utilizes magnetometer and barometer data. Assistance data (ephemeris, clock, navigation data bits, start position and GNSS time sync.) are received from a reference receiver via a TCP/IP connection.

When the receiver starts up, it gets a coarse time synchronization with an accuracy of around 30 ms via the data link (the NTP is used [10]). Via ZigBee, WiFi or the assistance link, it gets an approximate start position. Then it acquires via a long coherent integration GNSS signals. Tracking is started with independent channels. The GNSS NCOs utilize only the µ-trajectory velocity to adjust the NCO rate values (= *tight coupling*). The integration Kalman Filter (IKF) updates its state vector with GNSS code pseudoranges and WiFi/ZigBee positions and provides position corrections and accuracy values to the µ-trajectory generator. If the accuracy of the µ-trajectory and the clock error falls below a threshold value (e.g. 50 m for GPS C/A code), then the system switches into vector tracking (= *ultra-tight* coupling). Suitable state machines control the transition between tight/ultra-tight coupling and between tracking/acquisition of GNSS signals.

Modified GNSS Signal Correlation

The software receiver maintains an internal time scale that is defined by

$$t_\mu = t_{NTP} + \frac{\mu}{f_s}. \tag{1}$$

Here t_μ denotes the internal receiver time for the received IF signal sample with index μ. The symbol t_{NTP} denotes the start epoch derived from querying an NTP time server.

The partially coherent GNSS/INS scheme requires to modify the well known GNSS signal correlation formula according to

$$C = \sum_\mu s_{sat}(t_\mu)\, c(t_\mu - \tau)\, d(t_\mu - \tau) \exp\{-i\omega t_\mu\} \exp\{i\Delta\varphi(t_\mu)\}. \tag{2}$$

Here C denotes a generic correlator output (during acquisition or tracking), $s_{sat}(t_\mu)$ are the received signal samples, $c(t_\mu-\tau)$ are the internal PRN code replica samples (including the (C)BOC/BPSK modulation), $d(t_\mu-\tau)$ are the navigation message samples and ω is the internal Doppler frequency. The innovative term is represented by the predicted carrier phase $\Delta\varphi(t_\mu)$ that is used to cancel non-linear carrier phase variations during the correlation process. This term is absent for short coherent integration times.

The predicted carrier phase should match the true carrier phase $\Delta\varphi_{true}(t_\mu)$ of the received signal as close as possible. Deviations from the true carrier phase cause correlation losses being expressed as

$$L_{NL} = \left| \frac{1}{T_{coh}} \int_{t=t_0}^{t=t_0+T_{coh}} \exp\left\{ i\left(\Delta\varphi(t_\mu) - \Delta\varphi_{true}(t_\mu) - \omega_0 t_\mu - \varphi_0 \right) \right\} dt \right| \tag{3}$$

with

$$\omega_0, \varphi_0 = \\ = \arg\min_{\omega_0, \varphi_0} \int_{t=t_0}^{t=t_0+T_{coh}} \left(\Delta\varphi(t_\mu) - \Delta\varphi_{true}(t_\mu) - \omega_0 t_\mu - \varphi_0 \right)^2 dt \quad (4)$$

The later equation includes the used maximum likelihood estimation of the Doppler frequency. This estimation is carried out during acquisition and during tracking. The admissible Doppler range depends on the accuracy of the IKF clock drift state.

DATA SYNCHRONIZATION AND TIMELINE

Within the DINGPOS platform all sensor data is time-stamped with respect to the internal receiver time scale. IMU records and barometer measurements are directly time-stamped by the GNSS frontend (with +/- 2 µs accuracy), WiFi and ZigBee data is time-stamped by the software receiver (with sub-second accuracy).

To leverage the computational demands, we sub-divided the data processing in Fig. 7 into two main processing lines:

- **Real-time processing:** instant dead reckoning for µ-trajectory generation and real-time position output
- **Delayed processing:** Delayed dead reckoning with the IKF for error state estimation

The delayed process collects all measurements and estimates the position error and clock error, which is extrapolated in time and applied to the real-time processing. The delayed processing waits until all data arrives. The extrapolation of the state from the delayed processing to the real-time epoch can be done precisely, because the residual user dynamics (i.e. the dynamics not captured by the dead reckoning) is quite low. This scheme is depicted in Fig. 8. The y-axis 'data-time' denotes the reference epoch of the various measurements. Processing of data (e.g. signal correlation to convert IF samples into pseudoranges) shifts the data in the 'real-time' axis but leaves the 'data-time' unchanged. Step detection works almost instantly.

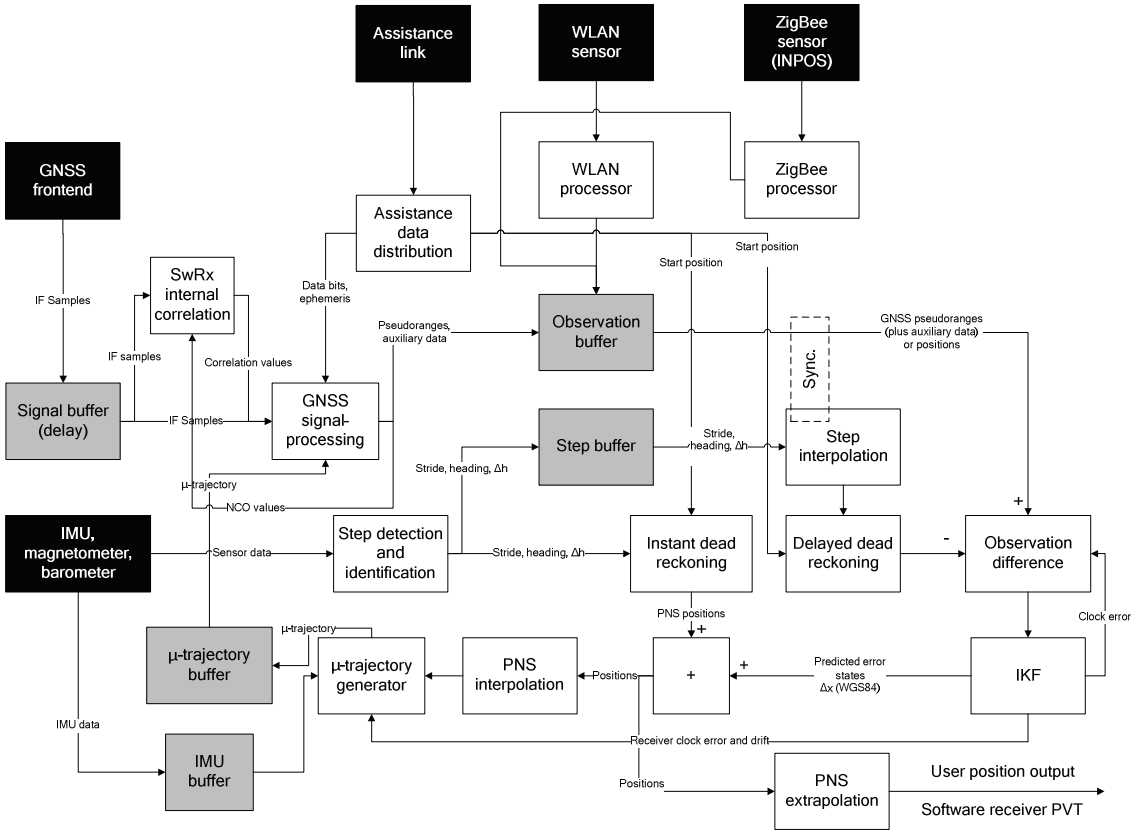

Fig. 7 *Detailed block diagram of DINGPOS GNSS/INS partially coherent ultra-tight integration algorithm*

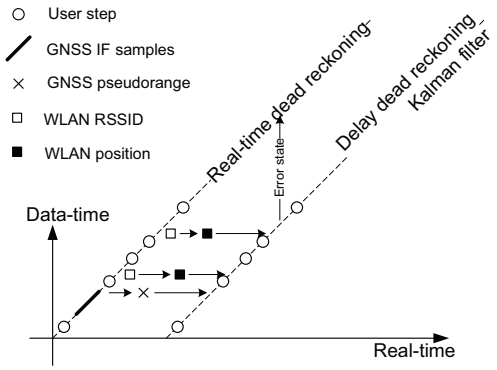

Fig. 8 *Data time-line of the DINGPOS processing*

KEY COMPONENTS

This section describes the key components of the DINGPOS prototype that are necessary to realize the long coherent integration times.

OCXO

It is a well known fact that the receiver clock error needs to be linear during the coherent integration time. Oscillator jitter or drift changes may otherwise corrupt the correlation results. A conservative bound

$$T_{coh}\sigma_A(T_{coh})f_{RF} \ll 0.5 \qquad (5)$$

of the oscillator Allan variance at T_{coh} = 10 s requires it to be much smaller than 3.1 x 10^{-11} [11]. The specification of the used OCXO of the NAVport 2 frontend provides a typical Allan variance at 10 s of 10^{-11}.

An explicit verification of correlation losses due to the OCXO jitter can be done be measuring the true carrier phase $\Delta\varphi_{true}(t_\mu)$ for a given GPS satellite at high SNR values. The test setup requires a static user position. The true carrier phase is computed as the unwrapped argument of the complex valued prompt correlator values. The true carrier phase is compared against a prediction based on previous IKF estimate of the clock drift. Segment-wisely detrended values of the true and predicted carrier phase are shown in Fig. 9.

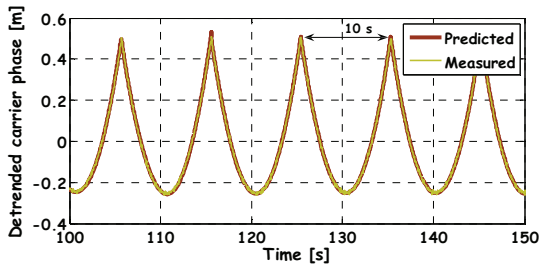

Fig. 9 *Predicted and measured GPS C/A carrier phase PRN29 for a static position*

The parabolic shape of the individual segments is caused by the satellite acceleration during the coherent integration interval of 10 s. We found that the OCXO is quite stable and a histogram of the losses in equation (3) over 250 s is shown in Fig. 10. Our experience is in concordance with the conjecture in [11] that (5) is a quite conservative bound.

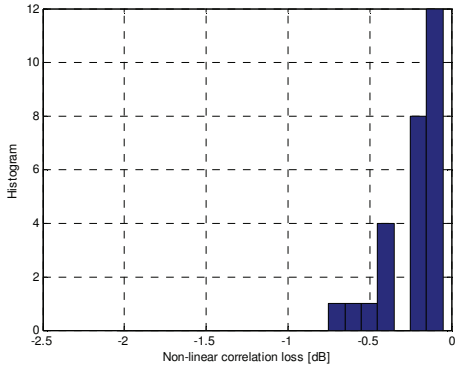

Fig. 10 *OCXO correlation losses over T_{coh} = 10 s*

Step Detection, Identification and µ-Trajectory

Equation (2) can be interpreted in a way that for a very long coherent signal integration a precise and long signal replica has to be provided to compensate the user motion effect on the carrier phase by calculating the phase term $\Delta\varphi(t_\mu)$. The user motion (µ-trajectory) is determined by a MEMS-type IMU. Since the strapdown-only accuracy of the µ-trajectory could not meet the required accuracy, a new approach combining dead-reckoning, an IKF and a modified strap-down INS has been designed and implemented. The system is described in detail in [1].

The accuracy of the µ-trajectory is demonstrated in Fig. 11. It shows a similar graph as Fig. 9, but this time a walking (t < 177 s) user is considered as well as a coherent integration interval of 2 s. Furthermore, a parabolic term (compensating the satellite motion) has been subtracted for better visualization.

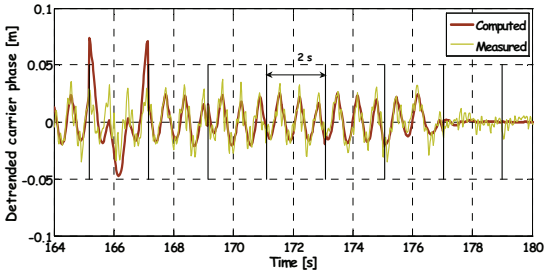

Fig. 11 *Predicted and measured GPS C/A carrier phase PRN30 for a walking user*

The proposed method works perfectly for a static user. It works well for a walking user and for high elevation satellites. For a walking user and lower elevations, the accuracy depends on the accuracy of the estimated heading. Occasionally, some intervals are affected by gross errors that cause larger correlation losses.

Assistance Data Link

Within DINGPOS project, the A-GNSS solution relies upon a Remote Procedure Call (RPC) client-server architecture, aiming:

- to exchange data between indoor (fixed) and outdoor (roving) platforms, strongly mitigating uncertainty on wireless link continuity
- to discipline operation and sharing of resources between tasks operating on the same host, like common memory areas, information storage queues, necessary for managing A-GNSS capabilities .

Open Network Computing (ONC)-RPC provides a good compromise between complexity, efficiency, real time operation and robustness, at the price of a very limited overhead on size of exchanged IP packets.

Both indoor and outdoor operating NavX receivers are supplemented with a custom RPC application layer and "RPC client" functions, through DLL components built on purpose for the DINPOS project (cf. Fig. 12).

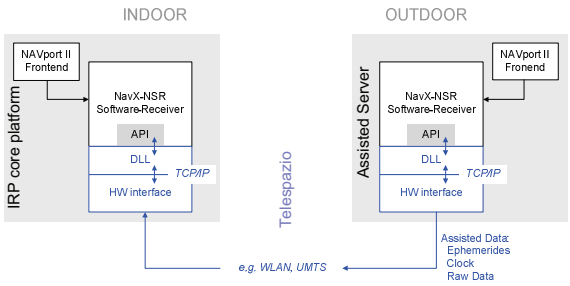

Fig. 12 *Assistance data transfer DLL*

RPC extends the capabilities of conventional software function (subroutine) calls, allowing to implement them "across a network" in the development of distributed computing systems. A "RPC call" is performed to invoke a function on a remote host, providing it necessary inputs and obtaining back processing results (cf. Fig. 13).

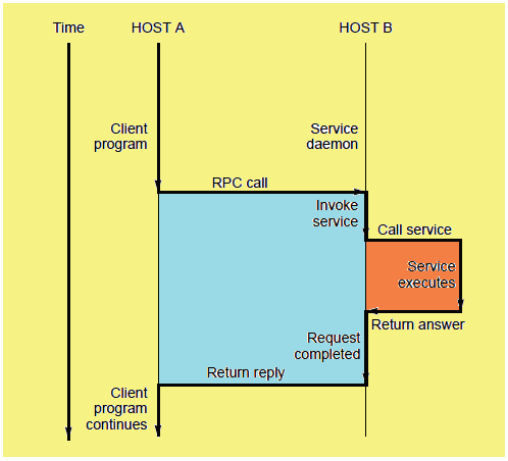

Fig. 13 *RPC client-server paradigm*

The capability resides upon use of IP, someway in a "transparent" way with respect to the application layer: in the main program, a RPC client "connects" first by selecting the networked host that will execute the specified routine, that is the so called "RPC server" side, and specifies the kind of transport: TCP or UDP over IP. After successful connection, the RPC client side will invoke available RPC server functions, each individually identified and characterized by own input-output parameters. The RPC paradigm allows the DINGPOS designer to concentrate on exchanged engineering parameters on both sides and not on all what pertains the access to a remote host resource.

Another advantage of RPC is that – on the application side – the hardware and the operation system of used platform is irrelevant for the programmer: before any data exchange, information is encoded in a binary format that is designed to be independent of the CPU register structure, which is named XDR (eXternal Data Representation).

Reference information about RPC and the related XDR model is mainly contained in Internet "Request For Comment" (RFC) public material: RFC 1831, RFC 1832, RFC 1833, RFC 2695.

The solution for DINGPOS A-GNSS architecture is actually based upon 4 (four) ONC-RPC servers, running as individual (and separate) background executable tasks:

- "svc.1": devoted to acquiring, storing in circular (LIFO) queues and distributing the GPS engineering parameters of the navigation message
- "svc.1.GAL": devoted to acquiring, storing in circular (LIFO) queues and distributing the Galileo engineering parameters of the navigation message
- "svc.2": devoted to acquiring, storing, transforming and distributing the rover position coordinates, as computed through GNSS, ZigBee subsystem or other means (e.g.: useful to provide "initial coordinates" to the rover platform)
- "svc.3": devoted to acquiring, storing in circular queues and distributing the raw bits of both the GPS and Galileo navigation message structure.

For instance, Fig. 14 depicts transfer of GPS/Galileo Navigation information encoded in an engineering format; note that RPC servers can be, in principle, activated on any host within the DINGPOS wireless LAN.

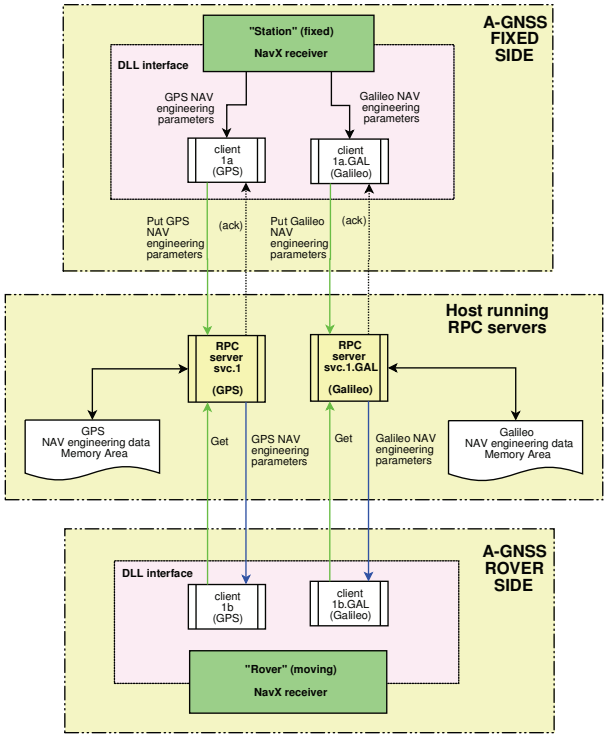

***Fig. 14** DINGPOS solution to exchange NAV message engineering information*

The NavX interface DLL implementation is such that in case of a wireless communication signal loss, RPC clients attempt to autonomously restore their connection with servers. Queues allow to mitigate asynchronous aspects in the operation between two sides (fixed, rover) and eventual temporary lack of digital communication exchange capability: critical information is never lost or corrupted and can be retrieved at later time instants, as soon as a new, successful digital connection is established.

WiFi Sensor

Since the positioning error of dead reckoning systems grows with time and distance traveled, aiding by position fix systems is common practice in navigation. GNSS is often severely degraded in indoor scenarios, hence other means of position updates have to be found. As means for providing proximity position in an indoor environment in which no GNSS signals are available WiFi can be used. Nowadays, in most buildings WiFi access points are installed which can be used as the necessary infrastructure.

WiFi can be used for positioning inside buildings basically by measurement of the received signal strength (RSS). Ranging of several APs like in the GNSS systems is not feasible today, since the precise time synchronization of the APs, like it is standard in the GNSS systems, is not implemented so far. This may change in the future. In this system WiFi shall be used as proximity indicator. The proximity is detected by observation of the RSS of the respective AP, but also the signal interference fading of the other APs in the direct vicinity of a (therefore very strong) access point.

As a prerequisite the APs used for positioning have to be registered to the system. Therefore a file containing the AP's MAC-address and the WGS84 position of the AP has to be provided. Also a calibration parameter for the individual transmission power can be added.

For determining the access point that is nearby the user and its notebook, the WiFi interface card of the notebook is used. The software receiver scans the visibility of all access points and provides their signal strength (RSSI, received signal strength indicator) in an interval of 2 to 3 seconds. Based on the MAC addresses and SSIDs (Service Set Identifier) the access points are uniquely identified and mapped to their known coordinates in WGS84. The proximity is determined by a system of several rules. Main parameter is the RSS of individual APs. For proximity to an individual AP, its RSS must exceed a predefined minimum. By adjusting this minimum, the detection range can be adjusted. Secondly, a significant RSS distance to neighboring APs must be measured, usually 2-4dB. As third rule the effect is exploited, that near strong access points the measured signal strength of other strong APs is decreasing. This RSS fading allows to use the derivative of the RSS values of neighboring APs to be used for proximity detections.

ZigBee Sensor and INPOS Sub-System

A ZigBee transponder system is inherited within the so called "INPOS" subsystem, developed outside the DINGPOS contract by internal R&D activities at Telespazio. It consists of a Crossbow IRIS "mote", integrated with a digital interface board MIB520. Total weight of the assembly is less that 40g without batteries (cf. Fig. 15).

***Fig. 15** "IRIS mote" (left) and MIB250 (right)*

The roving ZigBee device receives wireless signals from a number of fixed RF emitters of the same type; user position is then intended to be established knowing received power levels from each individual source. Propagation in an indoor environment is complex and cannot be easily modeled without measurements. Through a calibration procedure, which implies an operator to "visit" beforehand the indoor test area, radio-map layers of emitted signal power levels are recorded, so that a variation law for the received signal strength at roving sensor is obtained. This is the necessary input information for INPOS to be able to provide "user coordinates", which

are (natively) expressed as couples $\{ X_s, Y_s \}$, generated asynchronously w.r.t. the NavX GNSS receiver operation. $\{ X_s, Y_s \}$ are the coordinates of a point on a 2-dimensional map of the indoor test area that is intended to be navigated, given in units of "digital pixels" – thus, nothing to do with ECEF {x, y, z}.

The hybridization between ZigBee and GNSS positioning information is now envisaged for the DINPOS roving platform; it implies conversion from "map pixel" to ECEF coordinates. The adopted strategy for integrating different kinds of positioning information types is as follows. A "local plane" $\{ X_L, Y_L \}$ is to be properly selected in the indoor test area. It is defined to be orthogonal to geo-centre direction, with the ECEF origin at an arbitrary reference point P_{REF} , where:

$$P_{REF} \quad \Leftrightarrow \{ X_0 , Y_0 , Z_0 \}$$
$$\Leftrightarrow \{ \lambda_0 , \delta_0 , \rho_0 \} \text{ in geocentric longitude, declination and range coordinates.}$$

One moves from $\{ X_s, Y_s \}$ to such a "local plane" by applying a general linear transform of the form

$$\begin{bmatrix} X_L \\ Y_L \end{bmatrix} = \begin{bmatrix} X_{L0} \\ Y_{L0} \end{bmatrix} + \begin{bmatrix} \cos(\alpha) & \sin(\alpha) \\ -\sin(\alpha) & \cos(\alpha) \end{bmatrix} \cdot \begin{bmatrix} S_x & 0 \\ 0 & S_y \end{bmatrix} \cdot \begin{bmatrix} X_s \\ Y_s \end{bmatrix}. \quad (6)$$

Transformation parameters $\{ S_x, S_y, \alpha, X_{L0}, Y_{L0} \}$ are a-priori unknown. They are to be determined off-line, through an optimization tool built on purpose, given the map (pixel) coordinates of at least 3 points that are precisely geo-referenced in ECEF, using geodetic GNSS receivers in the test area. Transformation from local to ECEF is simply provided by a matrix, constructed from knowledge of P_{REF},

$$\begin{bmatrix} X_E \\ Y_E \\ Z_E \end{bmatrix} = \begin{Vmatrix} -\sin(\lambda_0) & -\cos(\lambda_0)\cdot\sin(\delta_0) & \cos(\lambda_0)\cdot\cos(\delta_0) \\ \cos(\lambda_0) & -\sin(\lambda_0)\cdot\sin(\delta_0) & \sin(\lambda_0)\cdot\cos(\delta_0) \\ 0 & \cos(\delta_0) & \sin(\delta_0) \end{Vmatrix} \cdot \begin{bmatrix} X_L \\ Y_L \\ Z_L \end{bmatrix}. \quad (7)$$

All previous formulae are applied by the RPC server "svc.2" which "stays in the middle" between NavX and the INPOS subsystem, to mediate all interface needs between non-homogeneous technologies. This server generates ECEF-converted INPOS information in the form of a NMEA alphanumeric encoding stream, so that logical interconnection results, allowing the desired hybridization of dissimilar sensors. This is depicted in Fig. 16.

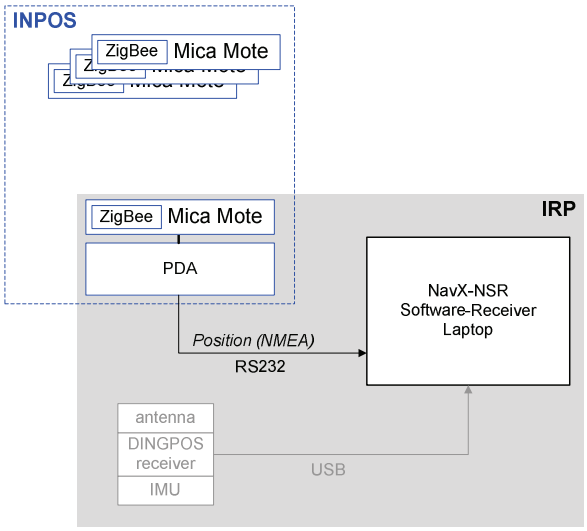

***Fig. 16** ZigBee use for DINGPOS*

Integrating Kalman Filter

The IKF is used to estimate the difference of the true-trajectory to the dead-reckoning trajectory and to estimate dead-reckoning biases as well as receiver clock parameters. The IKF state vector includes:

- Position error in WGS84 coordinates
- Heading bias and drift
- Stride bias
- Receiver clock error and drift

The IKF performs measurement updates using GNSS code pseudoranges and WGS84 position updates. The measurements are accompanied with appropriate accuracy information. The IKF detects outliers by computing predicted-minus-observed residuals and comparing them against a threshold.

GNSS Signal Processing

In order to realize the aspired long coherent integration via (2), the DINGPOS receiver makes in addition to the received IF signal samples the following information available to the signal processing:

- estimated user trajectory (position and velocity) plus estimation of user clock error and drift delivered from a μ-trajectory buffer
- satellite trajectory (position and velocity), satellite clock error and navigation data bits for each satellite in view delivered from an assistance data distribution

The providing blocks can be identified in Fig. 7.

For the generation of the reference signal during acquisition the course of the relative velocity between the satellite and the user during the integration interval is determined. Based on the information about the start time

of the received signal sample vector and the distance between the considered satellite and the user, a start spreading code phase for the reference signal is calculated (the start carrier phase is set to zero). In combination with the assumed clock drifts (mainly caused by the user clock) and taking into account the data bits, one subsequently derives the code and carrier phase time series, complex IF or the baseband reference signal, respectively, during the integration interval.

The correlation between the received and the reference signal is performed by applying a discrete Fourier transformation (DFT) and its inverse.

Non-coherent combination of coherent correlation results is applied, if necessary, to improve performance. Different Doppler frequency candidates are tested, mainly in order to cover the uncertainty about the user clock drift. After the correlation values are calculated for all Doppler frequency candidates, the maximum of C_j is determined within the assumed timing uncertainty interval and takes into account all Doppler frequency candidates.

In order to control the pair of false alarm and detection rate, the maximum peak is only accepted, if the ratio between the maximum and the average correlation value in the uncertainty interval exceeds a chosen threshold.

The system is designed such that signal processing has unambiguously to determine the timing error. However, the time interval, in which the maximum peak is searched, spans several data bits due to the assumed timing uncertainty ($\pm$ 30 ms). Consequently, it might comprise several prominent peaks depending on the combination of data bits. In the worst case of having only data bits with the same value, several peaks with the same value exist (if the noise is ignored). Therefore a procedure has to be established that calculates the auto-correlation function of the data bits within the considered signal interval and rejects the interval for acquisition, if the ratio of the peak at shift zero and the second largest peak is not larger than a certain threshold. This will be sufficient for signals that have one data bit per spreading codeword (e.g. Galileo/L1).

However in case of GPS C/A, one data bit comprises 20 spreading codewords, which leads to multiple only very slowly decreasing peaks in the vicinity of the peak at shift zero, even if the auto-correlation function of only the data bits fulfils the above described test; i.e., the bit clock synchronisation is likely derived with an error. In order to tackle this issue, it is foreseen to establish a permanent running test procedure, similar to the histogram method, during tracking that can identify the wrong synchronisation by using very long observation intervals and correct it.

During tracking internal optimised routines of the NavX-NSR receiver are used to generate the punctual and early-late correlation values for every spreading codeword. The 'internal correlation' block (refer to Fig. 7) receives an actual value of the code and carrier phase rate for each codeword. Both rates are determined based on the actual relative velocity between user and satellite (derived from the corresponding trajectory data) and clock drifts. In addition, the NCO code phase is set at the start of a long coherent integration interval in case of vector tracking. Its calculation is based on the corresponding pseudorange that is determined from the satellite and user µ-trajectory data, clock errors, and estimation of the ionosphere and troposphere error. Thus, the code phase can 'jump' at coherent integration interval borders, whereas the carrier phase is kept continuous.

The partial correlation results for each codeword in the integration interval are memorized and coherently combined to obtain the total result. The availability of the partial correlation results creates the opportunity to combine them alternatively assuming an additional frequency offset ω. Consequently, one can generate punctual correlation results as a function of an assumed residual frequency offset grid. Finally, one selects that punctual correlation value with the largest magnitude and assumes the corresponding frequency offset as actual frequency discriminator value (maximum likelihood Doppler estimation). This allows keeping long coherent accumulation intervals, although the accuracy of user trajectory might not be sufficient: deviations that lead to a constant frequency offset are detected.

At the end of an integration interval updated pseudoranges are calculated combining the pseudoranges at the start of the interval, the change of the geometric distance (based on user and satellite trajectory) during the interval, clock drifts, and the determined code and frequency phase discriminator outputs at the end of the interval. The updated pseudoranges are provided to the observation buffer (refer to Fig. 7) for further processing.

BENEFITS OF LONG COHERENT INTEGRATION

Coherent integration provides a number of advantages compared to a non-coherent integration. These benefits shall be outlined in the following sub-sections.

Multipath Mitigation in the Doppler Domain

The correlation process itself suppresses multipath signals, if the multipath Doppler frequency differs from the line-of-sight Doppler frequency. These phenomenon is called pre-correlation suppression and is for example described in [12]. A multipath signal is suppressed by

$$L_{pre}(f_D) = \mathrm{sinc}(\pi f_D T_{coh}) \cdot \tag{8}$$

where f_D is the Doppler difference of the line-of-sight signal to the multipath signal.

For an omni-directional antenna and a user moving with a speed v_{max}, the multipath probability expressed as a

function of the Doppler difference is given by the Jakes density

$$P(f_D) = \begin{cases} \dfrac{1}{\pi f_{D\max}\sqrt{1-\left(f_D / f_{D\max}\right)^2}} & \text{for } |f_D| < |f_{D\max}| \\ 0 & \text{otherwise} \end{cases} \quad (9)$$

where the maximum possible Doppler frequency f_{Dmax} is

$$f_{D\max} = f_0 \frac{2v_{\max}}{c}. \quad (10)$$

Fig. 17 shows the Jakes PSD (v_{max} = 1 m/s) and the product $L_{pre}(f_d)P(f_d)$. For coherent integration times larger than 20 ms, multipath is mitigated.

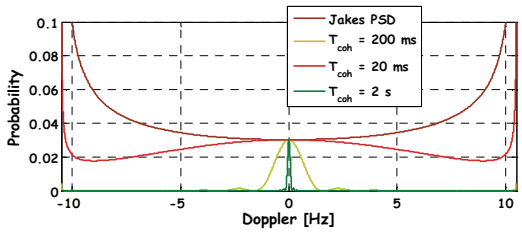

Fig. 17 *(Effective) multipath probability for various correlation times*

Cross-Correlation Protection by Data Wipe-Off

When we want to track a weak indoor signal, we have to face the problem that occasionally cross-correlation peaks of a strong signal (coming through e.g. a window) with the weak signal replica are larger than the desired auto-correlation peak.

The PRN codes itself provide a certain level of cross-correlation protection. This protection is enhanced when integrating over intervals longer than a data bit, assuming that the data bits from the two satellites differ from each other. Mathematically, the normalized cross-correlation is expressed as

$$R(\tau,\omega,d) = \left| \frac{\sum_\mu c_w(t_\mu - \tau) d_w(t_\mu - \tau) c_s(t_\mu) d_s(t_\mu) \exp\{i\omega t_\mu\}}{\sum_\mu c_w(t_\mu - \tau)^2 d_w(t_\mu - \tau)^2} \right| \quad (11)$$

where the indices s and w refer to the strong and weak signal. The cross-correlation depends on the relative code phase τ and Doppler ω of the two signals as well as on the given data bit configuration d.

For independent channel tracking, the channel of the weak signal may lock onto a cross-correlation peak. The protection (expressed as signal power difference) is

$$P_{Xcorr}(d) = \max_{\tau,\omega} R(\tau,\omega,d). \quad (12)$$

Averaging over all data bit configurations, the cross-correlation protection is obtained by

$$P_{Xcorr} = 20\log_{10}\left\langle P_{Xcorr}(d)\right\rangle_d. \quad (13)$$

Assuming the data bits being independent binary {+1, -1} and uniformly distributed random variables, an admissible code phase shift of [0…1023 chips] and an admissible Doppler difference of [-1 / T_{coh}, … ,1 / T_{coh}] yields the numerical values listed in Tab. 3 for the GPS C/A code signal.

Tab. 3 *PRN1/2 GPS C/A cross-correlation protection in independent channel mode*

Coherent integration time	Protection
20 ms	-24 dB
200 ms	-28 dB
2 s	-35 dB

A number of 500 data bit configurations was considered to obtain the values of Tab. 3.

Cross-Correlation Protection by Vector-Tracking

When the receiver is in the vector tracking mode, it cannot lock onto a signal cross-correlation peak because it is virtually impossible that the cross-correlation peak follows the desired auto-correlation peak.

As a consequence, the cross-correlation protection is enhanced by vector tracking. The cross-correlation acts as an additional noise to weak signal tracking and the cross-correlation noise power is

$$P_{Xcorr} = 10\log_{10}\left\langle R(\tau,\omega,d)^2\right\rangle_{d,\tau,\omega}. \quad (14)$$

Exemplary protection values are listed in Tab. 4. Here a value of 0 dB means that the cross-correlation noise power is identical to the auto-correlation power.

Tab. 4 *PRN1/2 GPS C/A cross-correlation protection in vector tracking mode*

Coherent integration time	Cross-correlation noise
20 ms	-42 dB
200 ms	-49 dB
2 s	-58 dB

Overall, we find that in vector tracking and with a 2 s coherent integration time, cross-correlation is perfectly mitigated.

Reduced Squaring Loss

It is well known that the accuracy σ_{obs} of a code pseudorange measurement (or a Doppler measurement) follows the generic formula

$$\sigma_{obs}^2 \sim \alpha \frac{B}{C/N_0}\left(1+\frac{\beta}{T_{coh}C/N_0}\right). \quad (15)$$

Here B is the tracking loop bandwidth. For low C/N0 values the squaring loss (the term in the parentheses) strongly increases. The squaring loss is partially mitigated by a longer coherent integration time.

PERFORMANCE ANALYSIS

This chapter contains results from two tests of the DINGPOS prototype.

NCS GPS/Galileo Signal Simulator

The performance of the DINGPOS system is best assessed in a simulated environment. We use an RF level GNSS signal simulator (the NavX-NCS) which is connected to the software receiver (cf. Fig. 18). The GNSS signal simulator is synchronized to a simulation of the dead reckoning, WiFi, ZigBeee and µ-trajectory sub-systems. Via the simulation environment, we have full control over all error sources and we can contrast the obtained results to results in a real environment.

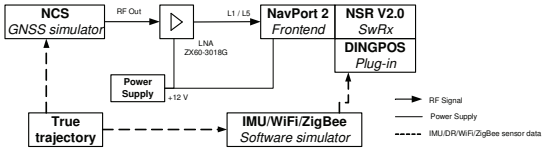

Fig. 18 *Test setup with the NCS RF GNSS signal simulator*

We simulate GPS L1/L5 signals together with GATE E1/E5a signals in basemode [13]. A skyplot of the transmitters is shown in Fig. 19 and the signal attenuation in Fig. 20.

The following results are all based on the GATE E1 (B+C) signal. For this signal an attenuation of 0 dB corresponds to a C/N0 value of 46.5 dBHz.

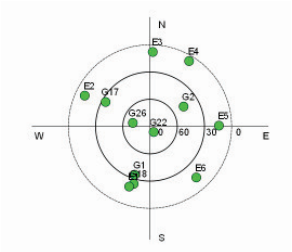

Fig. 19 *Simulated GPS/GATE constellation*

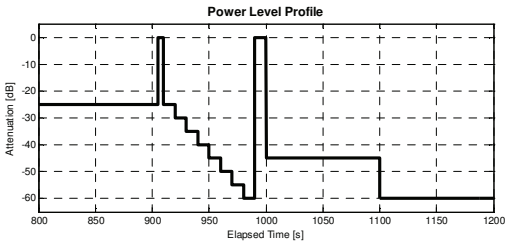

Fig. 20 *Simulated signal power profile (0 dB ~ 46.5 dBHz)*

The user trajectory is shown in Fig. 21 and Fig. 22. The trajectory runs through an area, where ZigBee updates are available and it passes by WiFi access points.

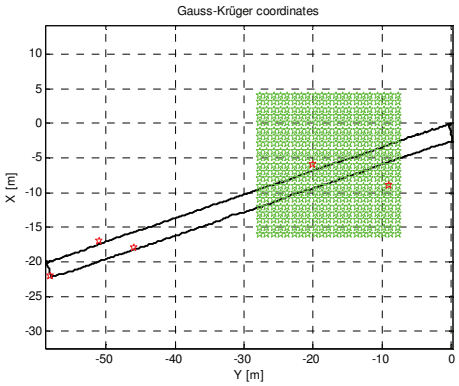

Fig. 21 *Simulated trajectory (red stars = WiFi access points, green points = ZigBee calibration points)*

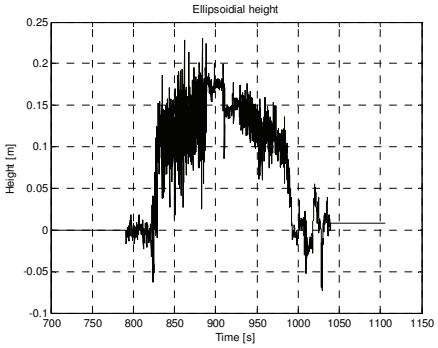

Fig. 22 *Height of simulated trajectory*

GATE E1 signals are acquired at a nominal C/N0 value of 21.5 dBHz with the settings of Tab. 5 and an exemplary correlation peak is shown in Fig. 23. The signal acquisition works very reliable, but is computational demanding (in terms of time and memory). This is currently being optimized.

Tab. 5 *Galileo E1 acquisition parameters*

Parameter	Value
Coherent accumulation length	0.17834 s
# of non-coherent combinations	2
Doppler bins	[2, 4, 6] Hz
Uncertainty interval	0.056 s
Sampling rate	8.274 Msps

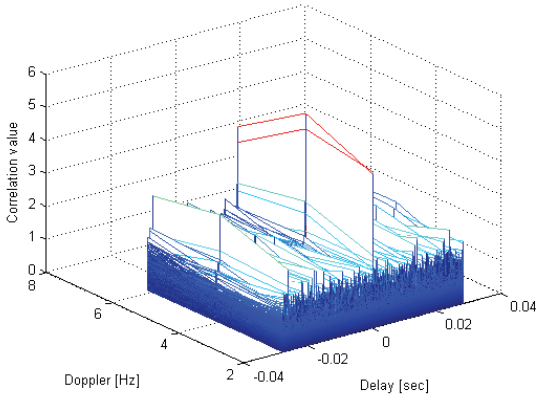

Fig. 23 *Correlation values vs. Doppler bins and uncertainty interval (E1, Sat PRN1)*

The estimated C/N0 during tracking is plotted in Fig. 24. A coherent integration time of 1 s was used. The receiver starts with independent channels and switches after t = 825 s to vector tracking. The switching causes a short term loss-of-lock, probably due to a jump in the receiver clock drift. This can be considered as a bug and is currently being investigated. However, the vector tracking is stable during this period (and during other periods of low C/N0).

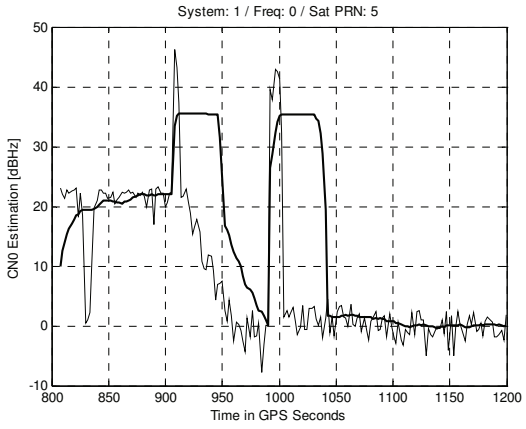

Fig. 24 *Estimated C/N0 during tracking (bold line is a 20 s running average of the thin line) (E1, Sat PRN5)*

The averaged C/N0 also nicely demonstrate the difference of a nominal C/N0 of 1.5 dBHz (t = 1000 – 1100 s) and if no signal is present (t = 1100 – 1200 s). This basically verifies that the system is able to track GATE E1 signals at 1.5 dBHz over 100 s. Further tests will indicate where the ultimate sensitivity of the system will be.

Finally, Fig. 25 shows the positioning errors. They depend largely on the simulated sensor characteristics (which are not outlined in this paper). However, Fig. 25 also nicely shows the positive effect of the ZigBee position updates bringing down the positioning error at t = 860 s.

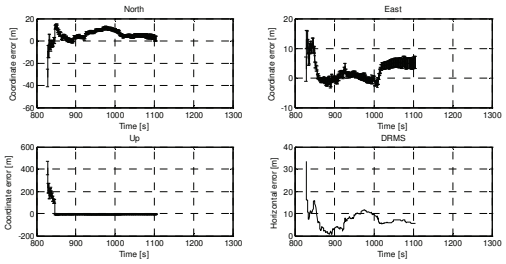

Fig. 25 *E1 positioning error for the simulated scenario*

Real World Tests

A second test is presented to characterize the (preliminary) performance with real GPS C/A code signals within the IFEN office building. The tests were partly carried on in a radiation shielded corridor causing extremely low C/N0 values. The test setup is depicted in Fig. 26 and relies on real GPS C/A code signals of Fig. 27 and real IMU and magnetometer data. WiFi, ZigBee or barometer data are not considered.

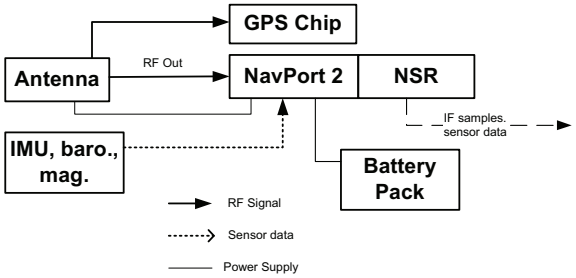

Fig. 26 *Test setup for the real GPS C/A test*

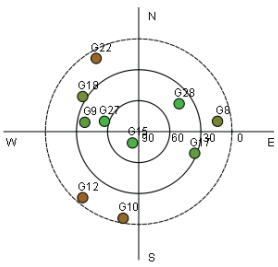

Fig. 27 *Sky plot for the real GPS C/A test*

For comparison, the GPS signal was also fed into a state-of-the-art high-sensitivity GPS chip that provided NMEA position output.

The DINGPOS signal tracking was performed with a coherent integration time of 1 s and narrow correlator (d=0.1 chip). The estimated C/N0 is shown in Fig. 28 for a single satellite in the independent channel mode or in Fig. 29 for all satellites in vector mode.

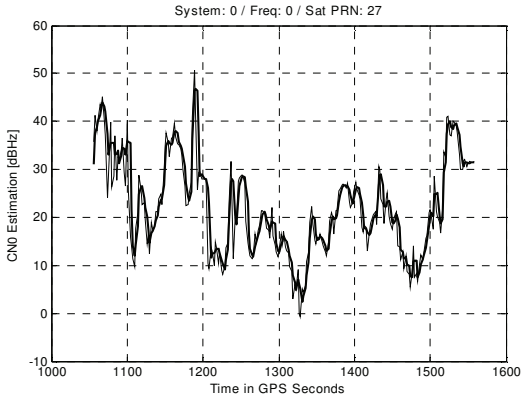

Fig. 28 *Estimated C/N0 in independent channel mode*

From Fig. 29 we see that at t = 1320 s, all but 2 signals fall below 10 dBHz which causes the (otherwise excellently performing) GPS chip receiver to stop positioning.

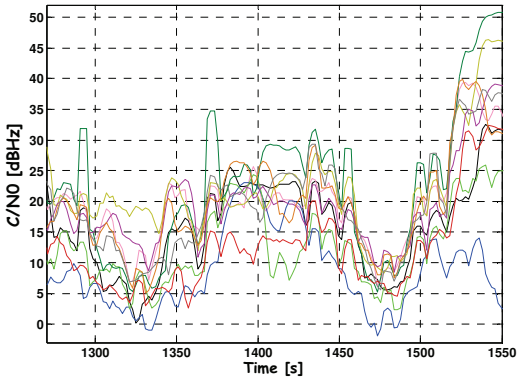

Fig. 29 *Estimated C/N0 in vector mode*

The obtained trajectories are shown in Fig. 30. We show the start point as circle, the end point as a rectangle. The data is taken from a longer test run. Thus, the start-points of the three trajectories do not coincide. The dead reckoning trajectory drifts away due to disturbances in the magnetic field. The combined solution corrects these disturbances, especially at the end where the signal power becomes high, because we leave the radiation shielded corridor and end up near a window.

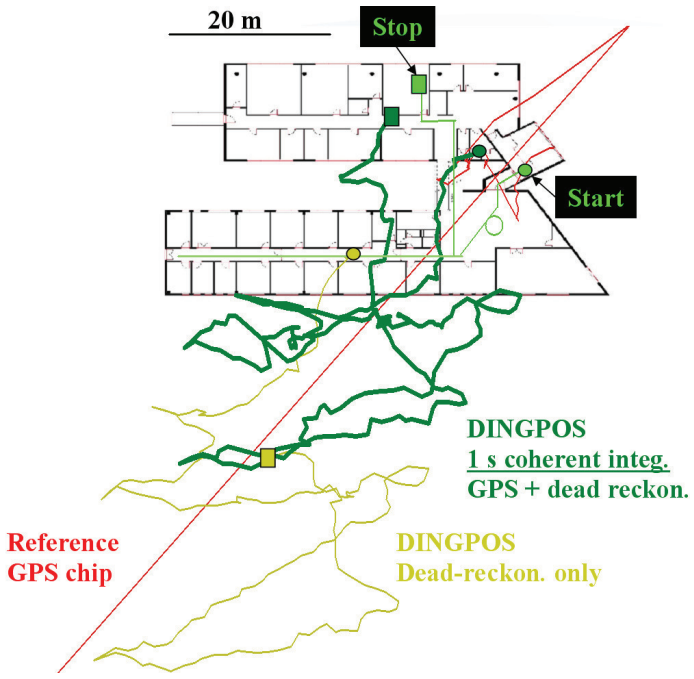

Fig. 30 *Real GPS C/A positioning results*

CONCLUSIONS

This paper has been written while the test readiness review of the DINGPOS project was ongoing. Performance optimization and final testing are still tasks to be done and all conclusions presented should be seen from this preliminary point of view.

So far, the system clearly demonstrated the ability to integrate coherently up to 2 s (dynamic) or up to 10 s (static) in the tracking mode. Signal acquisition can reach up to 250 ms coherent integration time, but this value is going to be increased in future. The simulations carried out with the signal simulator nicely verified the expected performance with the considered GPS/Galileo/GATE signals on L1=E1 and L5=E5a. Of special importance is the use of the L5 signals. Switching from a BPSK(1) signal to a BPSK(10) signal roughly corresponds to an equivalent C/N0 increase of 20 dB (no squaring loss) – 40 dB (with squaring loss) in terms of equivalent thermal noise ranging errors.

The probably most important element to achieve a good indoor positioning performance of this system is the dead-reckoning. GNSS pseudoranges may help to correct dead-reckoning errors by averaging over a long time. The test with the real GPS C/A code signal demonstrated that the system is able to track signals better than a state-of-the-art chip set. Furthermore, gross pseudorange errors (caused by cross-correlations or as loss-of-lock precursors) were virtually absent with vector tracking switched on and a coherent integration time of 1 s. However, the obtained positioning results do not satisfy our expectations and we will try to improve the system. Better magnetometer/gyro coupling and Kalman filter tuning might be important steps. Furthermore, the use of WiFi/ZigBee positioning in a real environment has still to be assessed. Map-matching is not considered in this project but is a very suitable tool to stabilize the user trajectory.

ACKNOWLEDGMENTS

The presented work was performed in the ESA funded contract: DINGPOS, ESTEC Ctr. No. 20834.

REFERENCES

[1] Niedermeier, H., et al., "Reproduction of User Motion and GNSS Signal Phase Signatures Using MEMS INS and a Pedestrian Navigation System for HS-GNSS Applications," *Proc. of the 4th ESA workshop on satellite navigation user equipment technologies, NAVITEC,* Noordwijk.

[2] Niedermeier, H., et al., "First results from supporting long coherent CDMA correlations by a MEMS INS and a Pedestrian Navigation System for HS-GNSS applications," *Proc. 6'th Workshop on Positioning, Navigation and Communication (WPNC'09),* Hannover, pp. 5-13.

[3] Winkel, J., et al., "DINGPOS - Taking Indoor GNSS Signal Processing to a Quantum Level," *Proc. of the 4th ESA workshop on satellite navigation user equipment technologies, NAVITEC,* Noordwijk.

[4] López-Salcedo, J. A., et al., "DINGPOS: A Hybrid Indoor Navigation Platform for GPS and GALILEO," *Proc. ION-GNSS 2008,* Savannah, pp. 1780-1791.

[5] Landis, D., et al., "A Deep Integration Estimator for Urban Ground Navigation," *PLANS 2006, IEEE/ION Position, Location and Navigation Symposium,* San Diego, pp. 927-932.

[6] Abbott, A. S. and Lillo, W. E., inventors. Global positioning systems and inertial measuring unit ultratight coupling method. USA. no. 6516021, 2003.

[7] Won, J.-H., Dötterböck, D., and Eissfeller, B., "Performance Comparison of Different Forms of Kalman Filter Approach for a Vector-Based GNSS Signal Tracking Loop," *Proc. ION-ITM 2009,* Savannah.

[8] Petovello, M. G., O'Driscoll, C., and Lachapelle, G., "Weak Signal Carrier Tracking Using Extended Coherent Integration With an Ultra-Tight GNSS/IMU Receiver," *Proc. ENC-GNSS 2008,* Toulouse.

[9] Sıçramaz Ayaz, A., Pany, T., and Eissfeller, B., "Performance of Assisted Acquisition of the L2CL Code in a Multi-Frequency Software Receiver," *Proc. ION-GNSS 2007,* Fort Worth, pp. 1830-1838.

[10] Wikipedia, "Network Time Protocol (NTP)," http://en.wikipedia.org/wiki/Network_Time_Protocol, 2008.

[11] López-Risueńo, G., et al., "User Clock Impact On High Sensitivity GNSS Receivers," *Proc. ENC-GNSS 2008,* Toulouse.

[12] Winkel, J. Ó. , *Modeling and Simulating GNSS Signal Structures and Receivers*. University of Federal Armed Forces Munich, Werner-Heisenberg-Weg 39, D-85577 Neubiberg, http://www.unibw.de/unibib/digibib/ediss/bauv, 2003.

[13] Heinrichs, G., et al., "First Outdoor Positioning Results with Real Galileo Signals by Using the German Galileo Test and Development Environment - GATE," *Proc. ION-GNSS 2007,* Fort Worth, pp. 1576-1587.