REAL-TIME TIME AND FREQUENCY TRANSFER USING GPS CARRIER PHASE OBSERVATIONS

Carsten Rieck, Per Jarlemark, Kenneth Jaldehag, and Jan Johansson
SP Swedish National Testing and Research Institute
Box 857, S-501 15 Borås, Sweden
Phone: +46 33 165440; Fax: +46 33 125038; E-mail: *carsten.rieck@sp.se*

Abstract

A client-server based data communication scheme was developed in order to establish a permanent and dynamic real-time GPS network for relative time and frequency transfer. The Kalman-filter-based real-time processor uses station pairwise common-view phase observations for estimating the receiver clock and tropospheric parameters. Orbit determination is based on real-time broadcast ephemerides and station coordinates are fixed and known. Real-time estimates were compared with clock solutions from postprocessed data resulting in standard deviation values in the order of 50 ps for short baselines.

INTRODUCTION

The time group at SP is in the process of decentralizing the base of the national time scale UTC (SP). By placing a number of high-quality clocks in different locations around the nation, a higher redundance is achieved. Clock combinations of these clocks with realizations at every location make distribution and availability of the national time scale more reliable. One major disadvantage is the loss of the link precision of an in-house time interval or phase measurement of the clock differences. In order to support a distributed time scale, a precise but cost-effective supplementing transfer link is needed. GNNS systems such as GPS are today widely used for geodetic relative positioning on the sub-centimeter level, which corresponds to the sub-100-ps level in time. It is commonly agreed that using geodetic postprocessing tools allow relative time comparisons in the order of 50 ps, provided that high-quality satellite orbits and a good modeling are used. Because similar technologies are applied as used by the geodetic tools, it is desirable to investigate how real-time implementations perform with all the restrictions they imply. Our requirements focus on the local (nationwide) usability, low computational overhead and robust/reliable performance.

During earlier investigations three different real-time methods were developed and initially performance tested [**1**]. The methods could be distinguished by how and if satellite clocks and orbits were estimated and whether differences were used or not. Even though the first measurements lacked reliable statistics, the *Single Baseline* approach using predicted orbits showed very promising results and was therefore chosen to be used for a study of a more permanent setup. The following sections will describe the developed system, discuss some results, and suggest improvements for the future.

METHOD AND EXPERIMENTAL SETUP

Sweden has a relative small number of primary clocks available. The majority of these clocks are placed on sites in southern Sweden that form part of SWEPOS© [**2**], the Swedish national permanent GPS network. It consist of 21 geodetic core locations and a similar number of secondary stations with lower requirements (see Figures 1 and 2). Some of these stations are also IGS tracking stations, namely ONSA, SPT0, KIR0, MAR6, and VIS0. The current setup for this study uses several different types of geodetic receivers: Ashtech Z12, Javad Legacy, and JPS E_GGD. Some SNR8000 were also used to provide real-time broadcast information. None of the receivers has a 1pps input option for time-coherent synchronization. Clocks involved in the measurements range from SWEPOS rubidium oscillators to several HP cesiums and two hydrogen masers located at Onsala Space Observatory (ONSA) and SP in Borås (SPT0). Except for the rubidiums, all clocks are routinely measured with either a time interval counter (TIC) measurement or with GPS-code common view/all in view against UTC(SP) and contribute to TAI.

Real-time methods are generally faced with the problem to deliver data to the processing in a reliable and synchronized manner. This task is sometimes underestimated and a weakness in the design of the communication part are cause to interruptions in the real-time processing. In order to help building a stable system, we decided to follow the basic design rule of separation of transport and application. This resulted in a modular system with modules for (**a**) receiver interaction/data extraction, (**b**) data communication, and (**c**) filtering.

(a) Raw Data Extraction Receiver Interface: DERI

A geodetic receiver connected to a primary frequency source to be monitored is the starting point of the data flow. The software module responsible for getting data from the receiver into the system has a number of functions to perform:

1. (serial) communication to the receiver, possibly on several channels
2. receiver initialization, health monitoring, sanity checking
3. data decoding, error check, archiving, data formatting, and time tagging
4. data interface to the communication software.

Figure 1: The SWEPOS core station locations, IGS ONSA and SPT0 at Onsala and Borås, respectively.

Figure 2: SWEPOS/IGS SPT0 in Borås, uses a temperature-stabilized antenna cable in a water bath.

All receiver models considered here offer real-time data output at variable sampling rate in one or several different formats. The problem of format variability was addressed by RINEX for non-real-time applications and for real-time by RTCM. But unfortunately, RTCM is not widely spread; besides, it has its limitations and it is not considered the ultimate interface for the real-time setup described here. It merely offers an alternative or generic way for data extraction from a receiver.

After a data stream is established, the DERI software has to decode and format the receiver data in a defined way. It has to detect and reject all invalid and corrupt data. From this point on, all data should be transparent to the communication software and of no importance at all. A time tag based on the receiver sampling epoch is added to the receiver data and allows distinction and data alignment of several data sources in the later stages of the data transfer. The DERI can be used to extract both observables, i.e. code and phase, and ephemerides from the receiver. It should also allow all relevant settings, e.g. oscillator behavior, to be controlled. The decoding software can with advantage be modularized for receiver specific data streams. The solution described here builds on decoding modules for RTCM and TurboASCII (AOA)

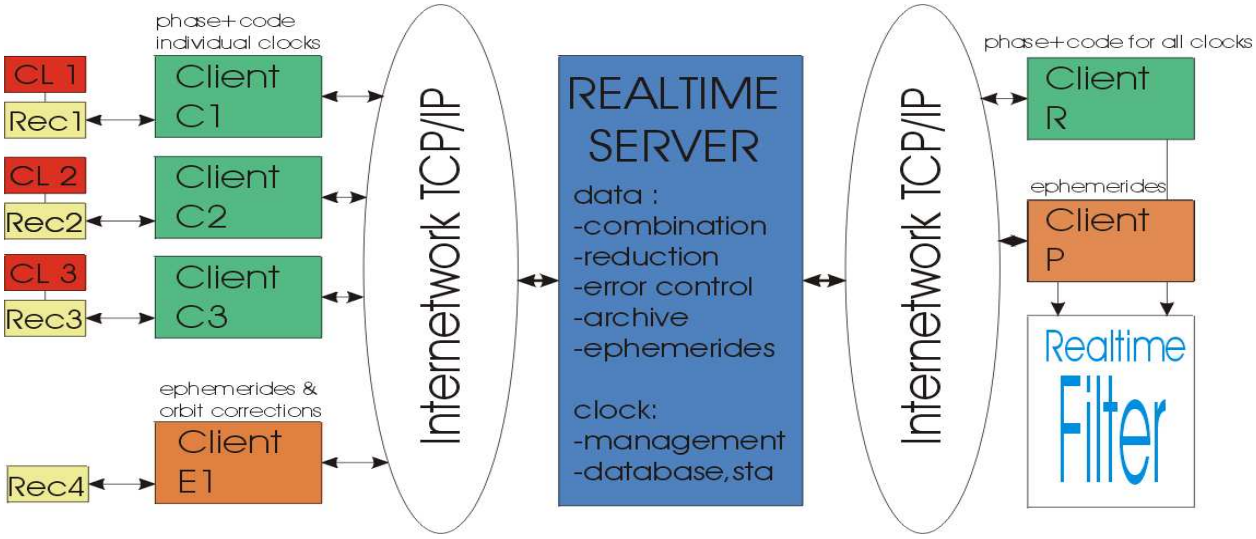

Figure 3: Real-time data transfer model,

real-time data streams. Support for native Ashtech and Javad data will be added in the near future. The data interface to the communication software is generally realized with UNIX pipes. Perl and C language was used to implement the receiver interaction and decoding. The software was compiled for a GNU/Linux environment.

(b) Real-Time Data Communication: RTDC

The real-time communication software was designed in a classical client-server architecture. Refer to Figure 3 for a depiction of the data flow from receiver to the processing. Three different communication parts can be distinguished:

1. a number of Sending Clients connecting to GPS receivers via DERInterfaces,
2. a single Server, and
3. two Receiving Clients for observables and satellite information, respectively.

An Internet network connects the clients and server together. The distinction between *server* and *receiving clients* was made for Internet security reasons in order to protect SP's internal computer network. Clients and server are naturally situated apart from each other on different machines, but of course it is possible and maybe sometimes desirable to place all parts of the system on the same physical machine. The implementation relies on TCP/IP Internet sockets available for all modern operating systems. Currently, everything is programmed using Perl 5.6+ under a GNU/Linux environment.

Sending clients are divided into two types: (a) observation data ($L1$, $L2$ and code) and (b) orbit information data (ephemerides). Both client types interface with the respective functions of the DERI for retrieval of the respective data type. The observation data are tagged with a clock identifier before it is send to the server. Furthermore, the

client has to establish and keep up the connection to the server, reestablishment of the connection might be necessary.

The *server* lives on a dedicated machine within a DMZ of SP's network. It is the central part of the communications system and its main purpose is to combine and synchronize the different data streams coming from the sending clients. A clock database holds all relevant information about the connected clients. This includes station coordinates, clock description, clock type and health, etc. New clocks are assigned a new clock identifier as soon as they connect and are accepted. Its clock and host data are added to the database. For efficiency a multi-process server design was chosen. It handles each incoming connection in its own sub-process. Inter-process communication is handled via shared memory. Two different data handlers can be distinguished: multiple receiving servers and two sending servers for observation data respectively ephemerides data. The combination of the data is realized on the sending server side with help of the time tags set in the DERInterfaces. The time information is stripped from the data before sending to the receiving clients is done. Ephemerides data have a given validity time and the combining server for orbit information classifies the data as valid/invalid. It sends one valid copy per satellite to the receiving ephemerides client. The observation time tag gives information about the GPS epoch the data was taken; the server allows a given latency time, otherwise it discards the data. Time source for the server is system time, which in turn is controlled by NTP using at least one primary NTP server.
Receiving clients are situated on the same internal SP machine as the real-time processing software. Both client types only receive the respective data and feed them further into named pipes connecting to the processing filter software. For the time being, the server only supports one single receiver client pair.

Post Real-Time Data Communication

For validation purposes, a post-real-time configuration was created. It allows feeding the real-time filter with observation and orbit information in the same manner as in the real-time case. Data source are RINEX observation files; orbit information, as well in RINEX format, are retrieved on demand from an NASA ftp server. The RINEX files are parsed and the respective data are synchronized, clock-id tagged and offered to the filter via a named pipe. The software can handle data requests from the filter for maximal throughput or can be run in a streamed mode with a given sample time. See Figure 4 for a depiction. Postprocessing was found useful for filter tuning and in cases where a real-time communication was not possible. Reprocessing of saved data has the advantage of being able to use a complete set of ephemerides, thus maximizing the number of usable satellite observations and in turn supporting a more robust clock solution.

Figure 4: Post-real-time data transfer model,

(c) Real-Time Kalman Filter: RTKF

The current filter software has evolved from a Kalman filter for real-time retrieval of the atmospheric zenith total delay (ZTD) as described in [**3**]. Following the main principles of the ZTD filter, differencing between observations of receiver pairs was introduced in order to eliminate the influence of the four, partly correlated, satellite position/clock parameters. This resulted in a modified Kalman filter with state variables describing (**a**) variations in the tropospheric delay, one per location, (**b**) differential receiver clock variations, one per receiver pair, and (**c**) initial phase states, one per receiver pair - satellite combination. A description of the Kalman filter principles is found in [**4**]. In the following some aspects of the modeling are described.

Station Position

The filter uses station coordinates presented in a model that corrects for the Earth crustal motions after the last glacial period as well as motion due to continental drift [**5**]. Furthermore, a model compensating for the main part of the elastic tidal response of the solid earth is used [**6**].

Satellite Position

Satellite positions are used for calculation of the geometrical distance between satellite and receiver antenna. The source for satellite position information are broadcast ephemerides, which can be combined with orbit correction information derived from Ultra Rapid Orbit Predictions as supplied by IGS [**7**]. Such corrections are currently not implemented. As for short baselines, the differential approach produces relatively precise differential distances; these differences degrade with increasing baselines due to geometrical errors introduced by incorrect satellite positions. Long baselines also suffer from “feed rotation” effects related to the polarization of the satellite signal. No compensation for these effects is currently implemented. The estimation of the correct receiver sampling time is crucial for obtaining accurate differential satellite – receiver

distances. Code observations are used to calculate the sampling times with an accuracy of about 0.1 μs, which introduces negligible differential delay errors for short baselines.

Ionospheric Delay

Phase delay variations due to the activity of the ionosphere are minimized by using the frequency dependent delay properties of the ionosphere by introducing the ionospheric "free" linear combination $L3$ [**8**]. The unmodeled higher order effects contribute with delay variations of at most a couple of ps.

Neutral Atmospheric (Tropospheric) Delay

The delay in the lower electrically neutral atmosphere (troposphere) can be divided into a dry or hydrostatic delay and a wet delay due to water vapor. The main part of the hydrostatic delay is compensated for using *a priori* zenith hydrostatic delay mapped to the observations using the Niell hydrostatic mapping function [**9**]. The Kalman filter model for the remaining atmospheric delay is zenith wet delay state variables, which are mapped to the observations using the wet version of the Niell mapping function. At the elevation cutoff angles of 15° used,the effects of mapping the remaining hydrostatic delay with the wet mapping function is considered insignificant. Near-zero baselines have highly correlated path delays. For robustness reasons, individual atmospheric delays are therefore estimated only if the spacial separation between the locations is greater than a design value, at present 1000 m. The states variables are modeled as random walk processes with variance rates of $2.25 \cdot 10^{-8}$ $\mathrm{m^2 s^{-1}}$.

Receiver Oscillator Variations

The estimation of receiver clocks is done in a somewhat reversed manner in order to support a stable and robust solution. In contrast to tropospheric delays, receiver clocks are expected to "jump" now and then, e.g. when a receiver resets its clock. This makes modeling difficult as to find appropriated variance rates for clock states. Furthermore, if high clock rates are allowed, then these variances mismatch the low variances set for the tropospheric parameters and would lead to ill-conditioning of the Kalman filter. A solution to this problem was found by studying the innovation vector, which represents all the non-systematic changes in the observation data. True clock changes contribute equally to all innovations independent of elevation. By averaging all innovations for a clock difference, a preliminary clock change solution is derived and removed from the observation vector before entering the Kalman filter. Typical innovations are now reduced to about 10 mm. Only innovations with a value smaller

than a design limit (at present 70 mm) are allowed to contribute to the preliminary clock change solution. This procedure maps certain amounts of tropospheric delay into the preliminary clock change solution. In essence, the remaining innovations contain the "true" variable tropospheric delays and the erroneous clocks. Thus, the time dependence of the clock states behave like tropospheric processes and have to be modeled accordingly. Such "tropospheric" clock differences are composed of changes in two different path delays; thus, these states are modeled with a random walk process and variance rates that are set to twice the rate for tropospheric delays, i.e. $4.5 \cdot 10^{-8}$ $m^2 s^{-1}$.

Phase Ambiguities

Initial phase measurement ambiguities are collected in state variables that are modeled as constants. The non-integer nature of these ambiguities, as a consequence of the use of L3, has no importance to this kind of real-time filtering. Initial phase states are reinitialized if the innovation filter described above discards certain satellite observations.

A step in the filter run can be summarized in the following sequence:

1. Update the current list of ephemerides and, if available, satellite position corrections using the ephemerides pipe, and sort out expired ephemerides
2. Read in new observations as sets of carrier phase observations ($L1$ and $L2$) and code observations by using the observation pipe
3. Calculate actual receiver sampling time by using code observations
4. Calculate the ionospheric free linear combination $L3$, from $L1$ and $L2$
5. Calculate all the individual distances from station locations to satellite positions at a receiver sampling time derived in 3
6. Calculate *a priori* hydrostatic zenith delay
7. Calculate the observation vector by differencing of $L3$ data from pairs of receivers with distances and *a priori* hydrostatic delays subtracted
8. Calculate a preliminary innovation vector
9. Calculate a preliminary clock change solution by averaging innovations from the same receiver pair
10. Subtract the preliminary clock change solution from the observation vector
11. Calculate an innovation vector

12. If largest innovation is greater than the innovation limit, then discard this observations from the preliminary clock change solution
13. Iterate to step 8 unless all innovations are within the limit

14. Subtract the preliminary clock change solution from each element of the observation vector

15. Compute the measurement covariance matrix as a combination of receiver noise, unmodeled receiver environment and unmodeled atmospheric effects that deviate from the mapped zenith delay as a function of the pointing directions[**10**]

16. Feed the observation vector to a regular Kalman filter calculation step, thus calculating the state vector and the error covariances

17. Add the preliminary clock change solution to the elements of the state vector in order to gain correct output

18. Iterate to 1.

RESULTS

Real-time filtering based on carrier phase observations is a relative method; thus, all results presented here have an arbitrary offset removed. Three different scenarios were considered: zero, short, and long baselines. The filter is considered to work well for short and zero baselines, whereas long baselines are known to suffer from the uncertainties in the orbit information used. All comparisons in the following are based on measurements against UTC(SP), which is a phase-stepped version of SP's CS5, an HP5071A (1642). UTC (SP) is connected to an Ashtech Z12 geodetic receiver.

In order to investigate the *Zero Baseline* behavior, 8 days in November 2003 were picked and fed to the filter software. SP's hydrogen maser HM1 (SigmaTau) provides a Javad Legacy GPS receiver with a 5 MHz signal. The receiver is connected in a true zero baseline to the antenna at the IGS marker SPT0 (see Figure 2). Figure 5 shows a typical output of the filter software. Comparing the phase of the real-time clock solution to a time interval measurement of the same clock pair yields RMS differences of about 100 ps; Figure 6 shows a typical day (MJD 52954, 315/2003). For the same day, Figure 7 depicts a comparison of the residuals of the real-time filter and the TIC residuals that were obtained after subtracting an individual drift of second order. Both residuals show similar features, which confirms the quality of the filter. Comparing the real-time solution to a clock difference that was calculated by a "geometric" zero-baseline GPS software, as used in [**11**], shows as expected a much better agreement, which is in the order 15 ps RMS. The discrepancy between the two methods is most likely explained by the similarity of the two GPS filters; both work differentially and for both cases the same input data were used. The TIC is an independent measurement method with independent noise characteristics, the counter PM6681 used alone has a

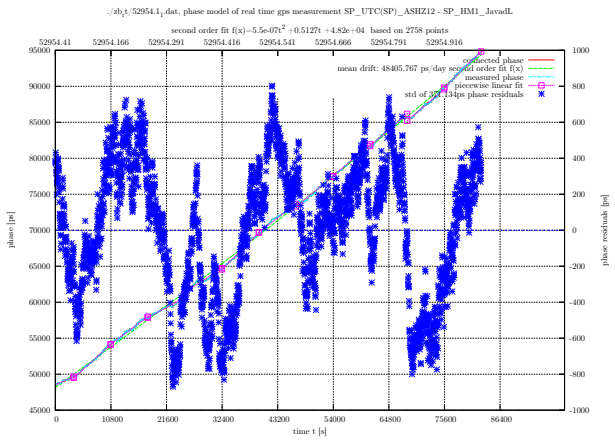

Figure 5: Output from the real-time filter for MJD 52954 for the GPS zero-baseline measurement SPT0−BOR0. The software allows to connection of phase-breaks due to ambiguity resets and calculation of model residuals after a quadratic drift is removed.

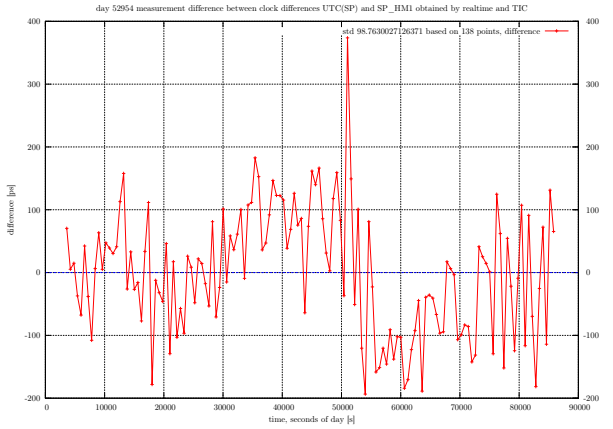

Figure 6: Difference between the clock differences for SPT0−BOR0 produced by the real-time filter and a TIC measurement. STDs of smaller than 100 ps are routinely obtainable. The graph represents data from MJD 52954 with the bias of the difference removed.

measurement uncertainty of about 50 ps. Nevertheless, the frequency stability of the three different methods is very similar, which in principle again confirms the filter functionality. See Figure 8 for a analysis of the data from day 314 till 321 year 2003.

An example for a *Short Baseline* is the clock comparison between Onsala Space Observatory (OSO) and Borås (SP) with a separation of about 68 km. The receiver used at OSO is an Ashtech Z12 that is fed with a 5 MHz signal from a hydrogen maser of type CH1-75A. Data from August 2003 were postprocessed with both the real-time filter and the geodetic GIPSY/OASIS II (PPP) [**12**] processing tools. Comparisons with the real-time solution was done in a similar way as in the zero-baseline case. RMS differences are as low as 30 ps per day and average around 50 ps for the month of August 2003. Even here, the comparison of the individual residuals show a high correlation between the two phases. Most of the periodic signature is expected to come from environmental changes in the involved receiver systems. Figures 9 and 10 show some results. The stability analysis in Figure 11 is somewhat difficult to interpret, since the Kalman filter (and also GIPSY's SQRIF) may artificially beautify the behavior of the observation data. Considering that the real-time filter does not allow real clock changes to propagate into the atmospheric parameters or into the residuals, its stability is only somewhat less than that of the far more complex geodetic tool. A GIPSY PPP "clock difference" is composed of two independent measurements; thus, the diagram in Figure 11 shows also a line of the $\frac{1}{\sqrt{2}}$ scaled Allan deviation of the GIPSY solution for the short term in an attempt to separate the two PPP tdp results (equally). The plot shows also the stability of the GPS-code common-view link used to report the OSO hydrogen maser to BIPM. For this study, the Allan deviations are

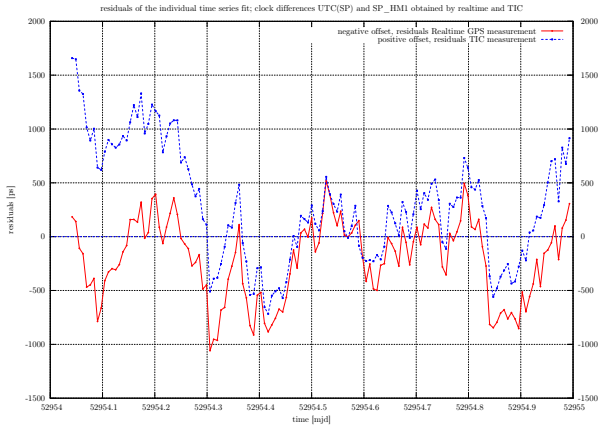

Figure 7: Comparison of the individual residuals obtained from clock difference data for MJD 52954 for UTC(SP) - SP_HM1. The upper curve represents the TIC measurement, the lower curve the residuals from the real-time filter clock solution. For clarity, the residuals are offset from each other.

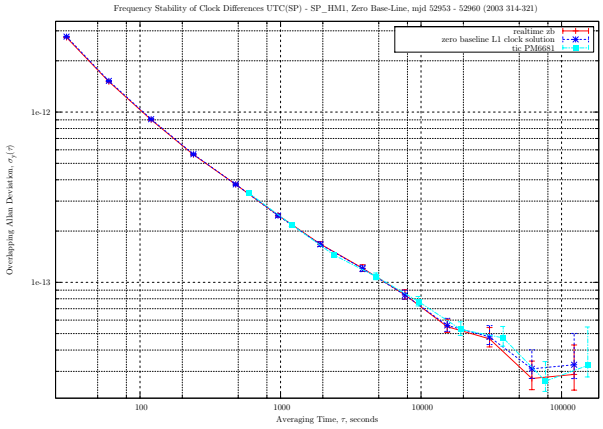

Figure 8: Frequency stability of the different methods used in a zero-baseline clock situation. All methods show a similar behavior. Source for the analysis is data from MJD 52953 till 52960. The feature between 20000 and 30000 s is due to the micro-phase stepper used to create UTC(SP).

ought to be used as quality indicators only.

Even though the filter was designed for local use, the performance for *Long Baselines* compared to a standard GIPSY PPP solution was investigated. Figures 13 and 14 show the results for the clock difference between UTC(SP) and a rubidium oscillator connected to a Javad GGD geodetic receiver in Kiruna (KIR0) in the north of Sweden. The resulting baseline is about 1200 km and the time period considered is again August 2003. RMS differences between GIPSY and the real-time filter results are well above 300 ps per day. This confirms somewhat the influence of the accuracy of satellite position coordinates on the performance of the real-time filter if one considers the individual GIPSY solutions to be equally precise. Even here the frequency stability of the links (plus clocks) has to be carefully analyzed. The real-time filter shows apparently the best performance, which is difficult to accept. One can argue, as one contributing factor, that the independent GIPSY PPP solutions for KIR0 and SPT0 use a different set of satellites with the respective noise introduced being more uncorrelated to each other than in cases with short baselines and more common satellite observations. The real-time filter, on the other hand, needs common observations to produce a clock solution. Nevertheless, the stability plot does not reveal any irregularities which would indicated problems in the real-time filtering. Figure 15 shows the development of the difference between the two methods for the long baseline for the first 10 days in August 2003. In contrast, Figure 16 plots the difference between the methods for the short baseline clock pair UTC (SP)−OSO_CH1-75A for the same time period. The “jumps”

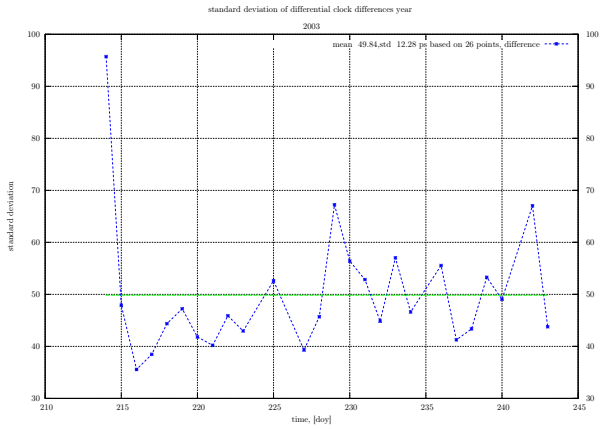

Figure 9: Daily STDs for the differences between RT and GIPSY for the short baseline SPT0-ONSA.

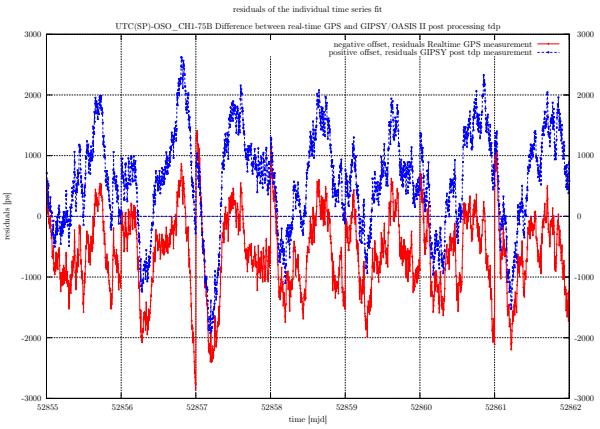

Figure 10: Comparison of the model residuals of RT and GIPSY for the short baseline SPT0-ONSA.

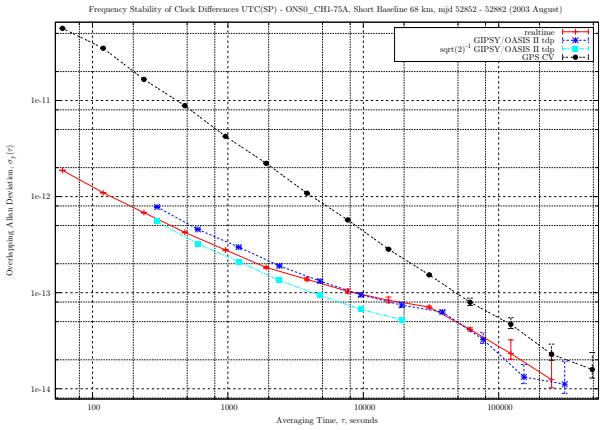

Figure 11: Stability of RT, GIPSY and GPS-code common view for the short baseline SPT0-ONSA.

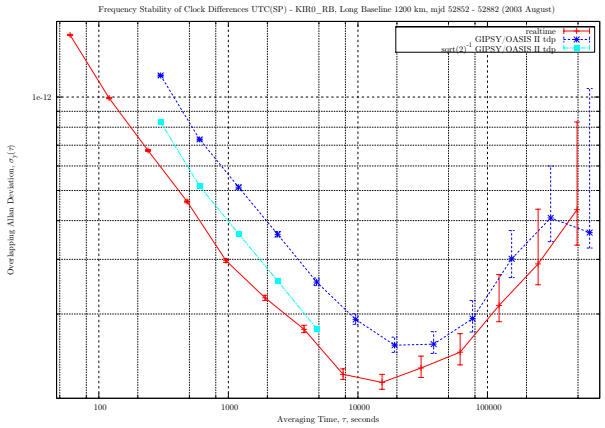

Figure 12: Stability of RT and GIPSY for the long baseline SPT0-KIR0.

in difference are due to day-boundary processing with GIPSY.

CONCLUSION AND FUTURE WORK

The study presented in this paper has shown the capability of the developed system to produce relative clock differences in real-time for short baselines that differ by 50 ps RMS per day compared to the differences produced by a standard GIPSY/OASIS II PPP solution. As a result of the current modeling, better clock estimates are produced for short baselines than for long baselines. The differential clock offsets can be used to produce relative frequency estimates of the clocks involved. The filter will be used as a permanent instrument to monitor the station clocks of the SWEPOS network and to support the formation of a distributed national time scale UTC (SP).

During the first period of operation and analysis of the data, several problems were pointed out and a future work plan includes following the points:

- conversion of the filter code from Matlab into a POSIX/GPL environment in order to improve the performance and the portability
- addition of support for more receiver data formats, e.g. native Ashtech and JPS
- support of dynamic insertion and removal of clocks into and from the filter
- implementation and use of satellite orbit correction information from IGS in order to offer a better support for long baselines
- improvement of the retrieval of broadcast information by using a dedicated code receiver
- testing and improvement of certain modeling aspects in the filter software e.g. feed rotation, improved tidal models, loading effects of ocean and atmosphere, influence of remaining hydrostatic delays in the wet mapping
- use of observations from lower elevation angles
- investigation and stabilization of the environment around the used receivers
- **establishment** of permanent operations and improvement of the operational reliability.

REFERENCES

[1] P. O. J. Jarlemark, K. Jaldehag, C. Rieck, and J. M. Johansson, 2003, *"First Results of Real-Time Time and Frequency Transfer Using GPS Code and Carrier Phase Observations,"* in Proceedings of the 2003 IEEE International FCS & PDA Exhibition Jointly with the 17th European Frequency and Time Forum (EFTF), 5-8 May

2003, Tampa, Florida, USA (IEEE Publication 03CH37409C), pp. 332-335.

[**2**] *http://swepos.lmv.lm.se/english/index.htm,* 2003.

[**3**] P. O. J. Jarlemark, J. M. Johansson, B. Stoew, and G. Elgered, 2002, *"Real time GPS data processing for regional atmospheric delay"*, **Geophysical Research Letters, 29,** No. 16, 10.1029/2001GL014568.

[**4**] R. G. Brown and P. Y. C. Hwang, 1997, **Introduction to Random Signals and Applied Kalman Filtering**, (John Wiley & Sons, New York).

[**5**] J. M. Johansson, et al., 2002, *"Continuous GPS measurements of postglacial adjustments in Fennoscandia, 1. Geodetic Results,* **Journal of Geophysical Research, 107,** B8.

[**6**] D. D. McCarthy (ed.), 1992, *"IERS Technical Notes 13,"* **IERS Standards** (Observatorie de Paris).

[**7**] *http://igscb.jpl.nasa.gov/components/prods.html, 2003.*

[**8**] B. Hofmann-Wellenhof, H. Lichtenegger, and J. Collins, 2001, **Global Positioning System, Theory and Practice,** fifth, revised edition (Springer, New York).

[**9**] A. E. Niell, 1996, *"Global mapping functions for the atmosphere delay at radio wavelengths,"* **Journal of Geophysical Research, 101,** 3227-3246.

[**10**] T. R. Emardson and P. O. J. Jarlemark, 1999, *"Atmospheric modelling in GPS analysis and its effect on the estimated geodetic parameters,"* **Journal of Geodesy, 73,** 322-331.

[**11**] C. Rieck, P. O. J. Jarlemark, K. Jaldehag, and J. Johansson, 2003, *"Thermal Influence on the Receiver Chain of GPS Carrier Phase Equipment for Time and Frequency Transfer,"* in Proceedings of the 2003 IEEE International FCS & PDA Exhibition Jointly with the 17th European Frequency and Time Forum (EFTF), 5-8 May 2003, Tampa, Florida, USA (IEEE Publication 03CH37409C), pp. 326-331.

[**12**] F. H. Webb and J. F. Zumberge, 1993, *"An Introduction to the GIPSY/OASIS-II"*, JPL Publication D-110088 (Jet Propulsion Laboratory, Pasadena).

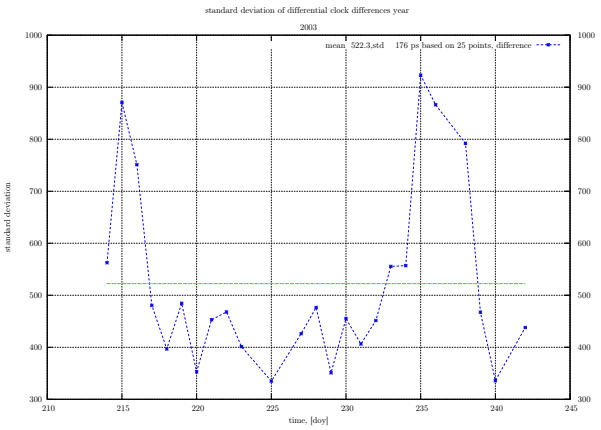

Figure 13: Daily STDs for the differences between RT and GIPSY for the long baseline SPT0-KIR0.

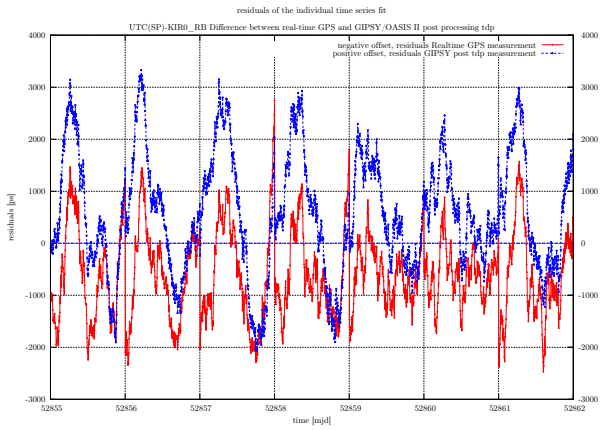

Figure 14: Comparison of the model residuals of RT and GIPSY for the long baseline SPT0-KIR0.

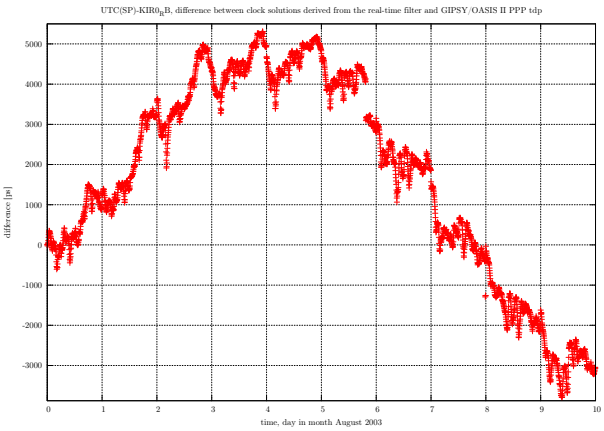

Figure 15: RT−GIPSY of the clock differences for the long baseline SPT0-KIR0, 10 days in Aug. 2003.

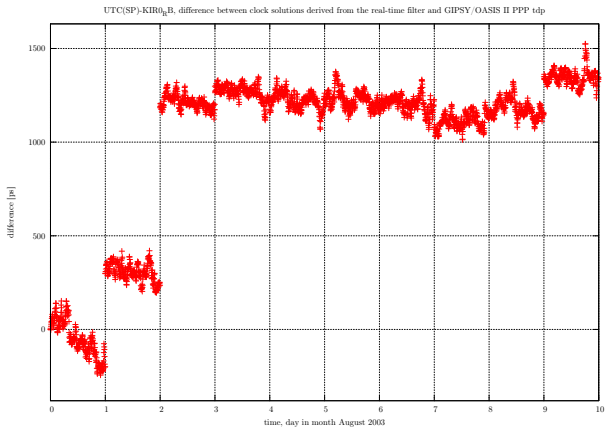

Figure 16: RT−GIPSY of the clock differences for the short baseline SPT0-ONSA, 10 days in Aug. 2003.

QUESTIONS AND ANSWERS

CHRISTINE HACKMAN (University of Colorado): I noticed in your modeling that you chose to model both the troposphere and the clock as random walk. Is there a particular reason you chose to model the clock as random walk, as opposed as to, like, white noise, as they often do in GIPSY?

CARSTEN RIECK: We think that is the better solution for modeling clocks. Those clocks also have different behaviors, but I think it is reasonable to model them with random walk.