Obtaining real-time sub-meter accuracy using a low cost GNSS device

Erwin Rademakers[1], Peter De Bakker[2], Christian Tiberius[2], Karel Janssen[1], Richard Kleihorst[1], Norddin El Ghouti[1]

***Abstract*—Autonomous vehicles require accurate position at all times in different environments at an affordable price. This accurate position can only be achieved when combining multiple positioning methods. One of these methods is presented in this paper: positioning based on a Global Navigation Satellite System (GNSS) to obtain absolute position. This solution should be at an affordable price with sub-meter position accuracy. At the University of Delft, the Netherlands, a low cost solution was developed in matlab for open areas which is called Single Frequency Precise Point Positioning (SF-PPP). It uses a low cost receiver with single frequency, single antenna and single GNSS constellation (GPS). The receiver provides raw measurements to the SF-PPP algorithm which corrects them for different kind of errors. This method was ported to a low cost Commercial Off-The-Shelf (COTS) embedded platform in C++. The selected platform is a Raspberry Pi version 2 with a u-Blox NEO 7P GPS receiver. The corrections for the raw measurements are received from a network service via a 4G modem. The PPP method is validated with an RTK system which is cm accurate. We evaluated the PPP method in different environments and conditions, with focus on open area, but also for harsh conditions on the highway and in an urban environment to know the current limitations of the method. For the open area environment a horizontal root mean square error (RMSe) of 0.5 m on position coordinates was achieved which fulfills our target of sub-meter accuracy. In harsh environments we suffer from reflections (caused by multipath receptions) and poor satellite availability due to obstructions from trees and buildings which makes the accuracy varying from 0.5 m up to 3 m. Future plans to improve the results involve using more satellites from other constellations like GLONASS, using the doppler shift to estimate the vehicle speed, using dual frequency receiver for ionosphere removal and closer integration with other low-cost sensors and vehicle model.**

I. Introduction

Autonomous vehicles require accurate position at all times in different environments at an affordable price. The positioning solution should be able to operate under harsh conditions like for example bad weather conditions: rain, fog and snow. It should operate in on-road and off-road environments where buildings, trees, bridges etc could make it difficult to obtain good positions. To be able to cope with all these environments and conditions a combination of different methods using different sensors is required.

[1]Flanders Make-Lommel Strategic Research Centre Manufacturing Industry Oude Diestersebaan 133, 3920 Lommel, Belgium erwin.rademakers@flandersmake.be

[2]Delft University of Technology Mekelweg 2, 2628 CD Delft, Nederland P.F.deBakker@tudelft.nl

Of course by using high cost sensors a high accurate position can be achieved. But such a positioning system should also be affordable, therefore low cost sensors are used.

In our solution shown on figure 1 we use the following sensors: a GNSS system, an IMU, camera and vehicle sensor information. The GNSS solution will provide absolute position. From the cameras we can determine the relative position from the visual beacons. The position of the detected visual beacon is retrieved from a map in order to determine the absolute position of the vehicle. From an Inertial Measurement Unit (IMU) and visual odometry method we can obtain relative positions. These methods provides positions in between the absolute positions methods. Sensor information from the vehicle in combination with the vehicle model are fused together with the absolute and relative positions to improve the positioning and to provide a position at all times. Besides the position, the system provides also an estimate of the speed and the attitude (pitch, roll, and yaw) of the vehicle. Further details and results of this multi-sensor fusion can be found in this paper [7].

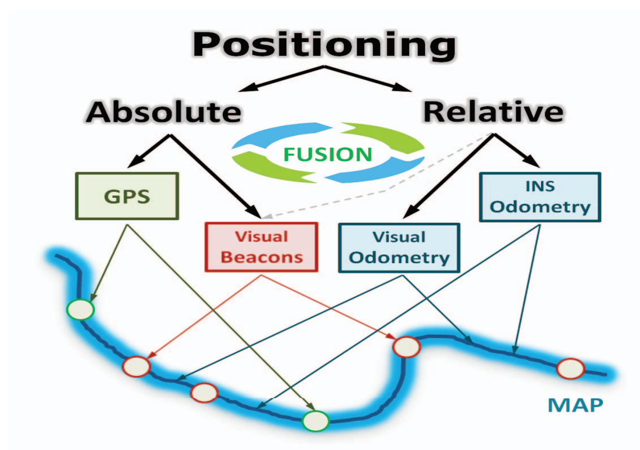

Fig. 1. Concept Positioning

The targets for this positioning solution is shown in the following table I.

	RMS Accuracy
Position	dm range
Pitch	0.10°
Roll	0.10°
Yaw	0.10°
Course	0.10°
Speed	0.1 m/s
Update rate	100 Hz
Latency	100 ms

TABLE I

The targets are set for vehicle speeds below 120 kph, here we take a typical speed on the highway.

The objective of this paper is the contribution of a low cost GNSS method to the fused position solution. A low-cost GPS solution can be found in smartphones, navigation systems, and its accuracy is rarely better than 5-10 m. On the other side of the spectrum, an RTK solution reaches cm accuracy but has a high cost, starting from 10k €. Here we implemented a position algorithm to enable accurate GNSS solutions for low-prices. Our target for the position accuracy in horizontal direction is set to sub-meter. And the solution should be constrained to a standalone small form factor and consist of Commercial Off-The-Shelf (COTS) devices. The solution should operate in real-time, with a target of 10 measurements per second. The 10 measurements per second are useful for high dynamic maneuvers of the vehicle, i.e. at high speeds and while turning.

II. Selection of the GNSS solution

Several GNSS solutions have been benchmarked regarding price and position accuracy. We benchmarked low cost receivers based on pseudo ranges, a smartphone using an Assisted GPS position (A-GPS) solution using Wi-Fi and cellular information. Also a receiver using the Satellite Navigation Augmentation System (SBAS) system to correct the position and using an advanced signal processing method called Precise Point Positioning (PPP) was benchmarked.

Figure 2 shows a global overview of the accuracies of the different GNSS solutions, which is an adapted version from ESA navipedia [9]:

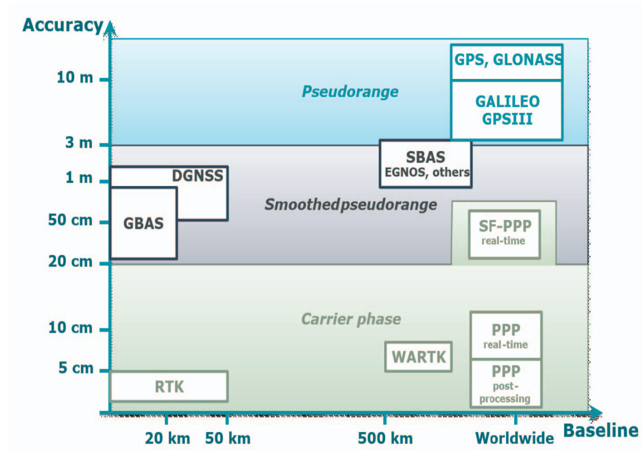

Fig. 2. Accuracies of different navigation solutions

Besides this global overview of the position accuracies also some tests were conducted on some GNSS solutions to see the position accuracies of these solutions.

Here we checked the position accuracies of the receivers against a reference system (see table II). Specification of this reference system see later in the experimental setup.

The used receivers are:

- smartphone iphone 6 plus, A-GPS
- Ublox 7p, only GPS (SBAS)
- Sirfstar III, low cost gps
- TU Delft SF-PPP matlab version, using the ublox 7p

All receivers are mounted on the roof except for the iphone, this was mounted behind the windscreen and so not all satellites could be received.

Receiver	North $\overline{RMSe}$ (m)	East $\overline{RMSe}$ (m)	$\overline{GDOP}$	$\overline{Sats}$
SF-PPP	0.86	0.33	2.00	9.96
Ublox 7p	1.24	0.29	0.98	10.00
Iphone6	7.17	13.69	-	8.00
SirfstarIII	3.92	5.73	0.82	10.00

TABLE II

As a result of the benchmarking, Precise Point Positioning has the best performance towards price and position accuracy, many research results also point in this direction. The PPP algorithm proves to be more accurate than the SBAS system since the SBAS system only corrects fewer error sources and the corrections of the SBAS system are less accurate.

III. Single Frequency Precise Point Positioning

A specific solution of the PPP method has been developed in cooperation with Delft University of Technology, The Netherlands [5] [3] [8]. Here a single frequency, antenna and GPS constellation receiver is used which is capable to provide raw measurements. These raw measurements are then corrected for all possible errors, like errors caused by ionosphere and troposphere, errors in satellite clock and orbit (see Fig. 3).

These errors are decomposed into the following table with its potential improvements on the raw measurements (range):

Error Component	Description	Potential Improvement
Ionosphere	Free electrons (sun) influence satellite signal (electromagnetic wave), Klobuchar model used, above > 1 km	7 m
Troposphere	Temperature, humidity, pressure influence, below < 1 km, Saastamoinen model used	2.5 m
Ephemeris data	Satellite data (nav message)	1 m
Satellite clock drift	Clock drift (even though atomic clock is used)	1.5 m
Dcb	Differential code bias, satellite hardware delay	50 cm
Phase windUp	Rotation of the antenna for both receiver and satellite.	dm
Sagnac effect	Earth rotation during signal travel.	Up to 30 m
ROA	Satellite orbit correction (radial, allong, crossing direction)	Up to 10 cm
Relativistic effect	Relativistic clock correction, a fast moving clock runs slower	Up to 21 m
Solid Earth Tides	Tides: Moon – earth interaction	5 cm horizontal and 30 cm vertical (in position)

Fig. 3. Error components

The data for a number of these corrections are retrieved over the internet from a network service. In real-time it will use predicted corrections for the ionosphere. Using a stochastic model an improved position is computed. Note that the network service enabling is still under development and is part of a large scale research project, so it is not yet a completely stable service. It is free of charge.

A. The receiver

A receiver is required which can be configured such that it will bypass its embedded processor (positioning algorithm) and delivering raw data of pseudo-range and phase measurements instead of the computed positions (in the form of National Marine Electronics Association (NMEA) messages). It allows us to do our own signal processing on the raw data (see Fig. 4).

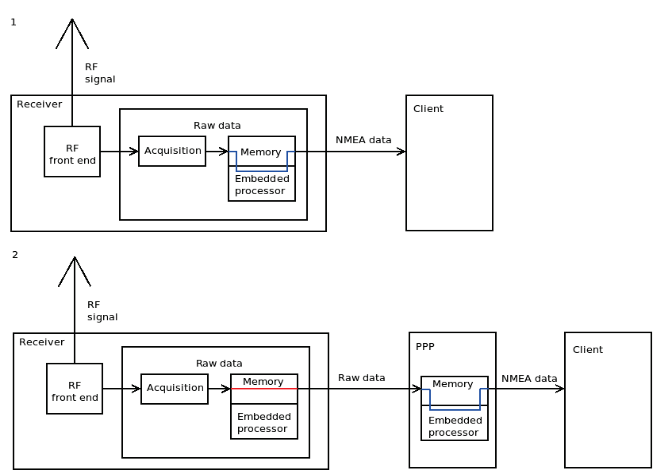

Fig. 4. Receiver modes: 1: Normal mode 2: Bypass mode

B. Model

The SF-PPP method is from a mathematical point of view an optimal position estimation with integral processing of pseudo range and carrier phase measurements together. The observation equations of this SF-PPP model are given for the pseudo range and carrier phase (which are non-linear).

$$P = \rho + c(T_{rec} - T_{sat}) + d_{orbit} + d_{tropo} + d_{ion} + d_{dcb} + \varepsilon_P \quad (1)$$
$$\phi = \rho + c(T_{rec} - T_{sat}) + d_{orbit} + d_{tropo} - d_{ion} + d_{windup} + \lambda N + \varepsilon_\phi \quad (2)$$

where
P is the pseudo range on L1 (m);
ϕ is the carrier phase on L1 (m);
ρ is the true geometric range (m);
c is the speed of light (m/s);
dT_{rec} is the receiver clock error (s);
d_{sat} is the satellite clock error (s);
d_{orbit} is the satellite orbit error (m);
d_{tropo} is the tropospheric delay (m);
d_{ion} is the ionospheric delay on L1 (m);
d_{dcb} is the differential code bias (m);
d_{windup} delay due to the rotation of the antennas (m);
N is the phase ambiguity on L1 (cycles);
λ is the wavelength on L1 (m/cycle);
ε_P is the noise in the pseudo range equation (m);
ε_ϕ is the noise in the carrier phase equation (m);

The observations are the raw measurements of the pseudo-range and carrier phase with noise. The pseudo ranges provide the distance to the satellites. The same counts for the phase measurements but expressed in the number of cycles at the given frequency.

The unknown parameters of the model are the three position coordinates in range ρ and the receiver clock offset T_{rec}. The unknown parameter for the observations is the phase bias since the carrier phase is ambiguous. These unknown parameters are solved using a recursive least square solution. First a linearization technique is used, then through iteration the position is obtained. The details about the linearization can be found in [4]. The weights of the least square solution are defined by the standard deviations of the expected error components and the elevation angle of the satellites. Receiver Autonomous Integrity Monitoring (RAIM) is used to assess the integrity of the raw measurements. When faults (outliers) are detected by the RAIM process, the bad measurements are removed and a new processing is started until no more faults are detected, or no more than 4 raw pseudo-range measurements (satellites) remain. The estimate of the ambiguity parameter from the previous epoch is used in the current epoch to further improve the accuracy of the position.

C. Architecture

An analysis and design was performed on the SF-PPP matlab implementation by Delft University of Technology, the Netherlands. At Flanders Make a C++ implementation was made on an embedded environment. The code runs on a Raspberry Pi version 2 (RPi v2). For the GPS receiver a u-Blox NEO 7P Evaluation Kit (EVK) was used with a patch antenna. Corrections from the network service are obtained using a 4G modem.

In figure 5 the used hardware is presented.

Fig. 5. Hardware components: 1: RPi v2, 2: U-Blox 7P EVK, 3: Battery, 4: 4G modem

A battery pack can be used in case no other power supply is available.

The SF-PPP positioning algorithm was developed on top of the open source software BNC [6]. The ublox driver software from RTKLIB [10] was ported to the BNC software to retrieve the raw measurements. The deployment diagram and the different components are shown in figure 6.

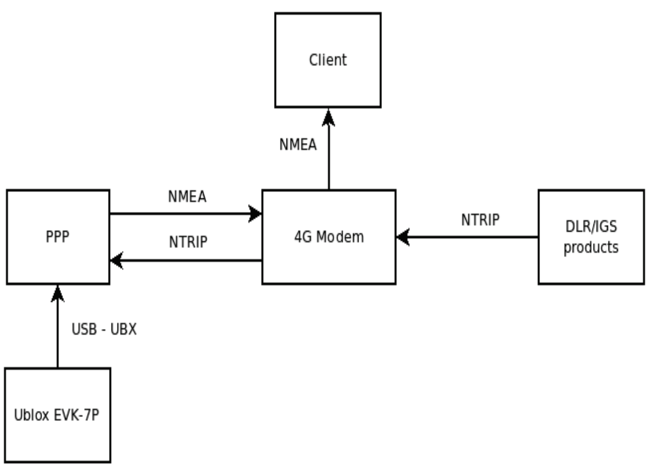

Fig. 6. System overview

The u-Blox NEO 7P receives the GPS L1 signal from the satellites. In the software on a thread the raw measurements are read from the receiver using a serial interface. The data is in UBX format, a dedicated ublox data format. At the same time in another thread the corrections for the error components and the satellite navigation data (ephemerides) are received from a network service, also called a caster. The protocol of the network service is Networked Transport of RTCM via Internet Protocol (NTRIP). The data format of the navigation messages is the Receiver Independent Exchange Format (RINEX). From the satellite navigation data, also called the ephemerides, the position of the satellites are computed. In the receiver component the raw measurements are corrected. The raw measurements as well as the corrections are recorded on disc to be able to post-process them later. In the PPP model the receiver position is computed by solving the equations 1 and 2 for all satellites together using a recursive weighted least square method. The computed position is made available on a User Datagram Protocol (UDP) network stream in the NMEA format. The NMEA message consists of the GPGGA, GPRMC and GPGST of the solution. Basically they contain the time, the latitude, longitude, height, DOP and the variances of the computed solution.

The total cost of a standalone system is 250 €, taken into account that the network service for the corrections are for free.

A breakdown of the cost can be found in table III.

Devices	Cost
RPi v2 (with enclosure)	50 €
Ublox 7p	90 € (1-9 pieces)
4G dongle	50 €
Battery	50 €
4G data connection	10 €/month

TABLE III

To retrieve the corrections from the network service a 4G modem is used. The data traffic from these network services is rather low, in the range of kilobytes for a few seconds. The corrections are updated every 5 seconds. We used the corrections from the German Aerospace Center (DLR, German: Deutsches Zentrum für Luft- und Raumfahrt e.V.), but the IGS RTS provides similar products.

IV. Experimental setup

In this section the setup of the experiments are presented: the used vehicle (see Fig. 7) equipped with the GNSS receivers, the used reference system and also the different environments used for the experiments. The antenna position offsets where measured in order to corrected and compare it against the reference system. The influences of the antenna (cross talk), surface of the roof of the vehicle is unknown.

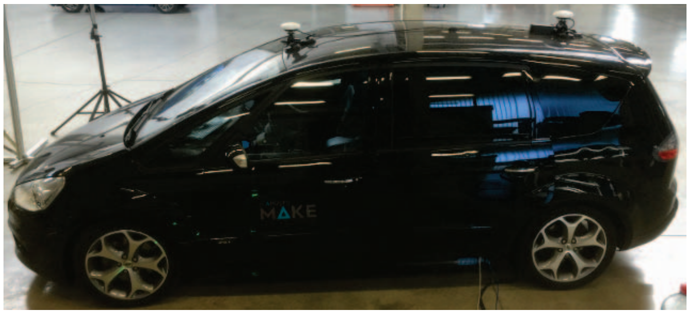

Fig. 7. Flanders Make experimental vehicle

We evaluated our PPP algorithm and system in different environments and conditions, in an open area (see Fig. 8), on a highway (obstructed by trees) (see Fig. 9) an urban environment (obstructed by buildings) (see Fig. 10). The tests were repeated a number of times.

Fig. 8. Open area (Google Maps, 2015 [1])

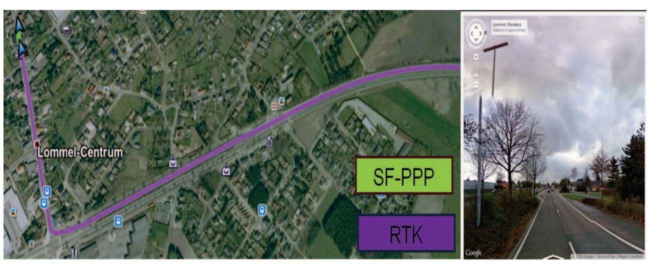

Fig. 9. Highway (Google Maps, 2015 [1])

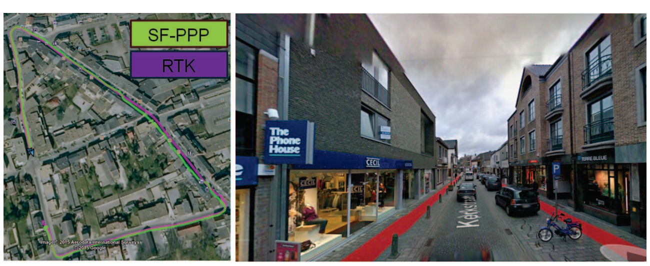

Fig. 10. Urban (Google Maps, 2015 [1])

The used reference system was the Ekinox model E from SBG (ground truth). This system combines RTK GNSS and an Inertial measurement unit, running an enhanced extended Kalman filter (EKF) at 200 Hz which fuses in real-time inertial data with RTK GNSS positions for the best performance. It uses 2 antennas. The antenna at the back of the car is to determine the absolute position and the antenna on the front of the car is to determine the heading of the vehicle. The IMU is mounted at center of gravity (COG) of the vehicle. The accuracy of our reference system was verified at know reference points [2]. The mounting offset between the reference system and the SF-PPP is compensated when computing the RMSe. In open areas the reference system is running in RTK internal mode which means that the best possible position accuracies of 2 cm can be achieved. In harsh conditions the reference system is not always able to stay in RTK internal mode and deviates from the real ground truth and so might impact the computed RMSe of SF-PPP.

V. RESULTS AND DISCUSSION

First the results are presented for the different environments. Next some more statistics are presented in an open area where satellite availability could influence the results. At the end the real time computation time performance and delays of the SF-PPP method are presented.
The position accuracies are computed based on the North-East-Up (NEU) coordinate system. For the North and East direction the RMSe and Geometric Dilution Of Precision (GDOP) is computed. Here GDOP is computed from the used satellites (RAIM).

A. Different environments

The results for the different environments are presented in this section. The tests were performed under the favourable conditions regarding available satellites and GDOP for that time of the day under normal weather conditions (no fog, no rain). The tests are repeated a number of times.

1) Open area: In an open area we can rely on all available satellites in the sky, no obstructions from trees or buildings.

Figure 11 shows the error in the North and East direction, the number of satellites and GDOP value of one of the experiments performed.

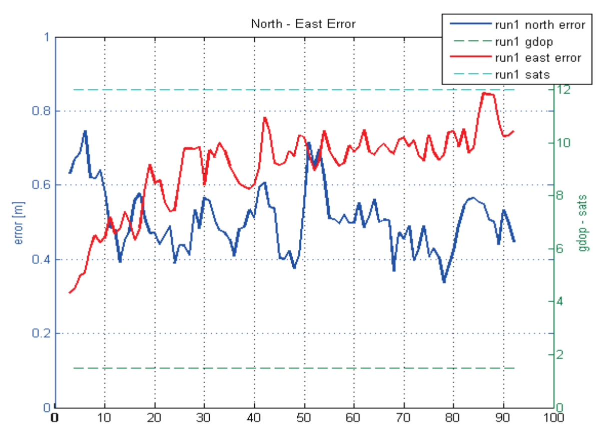

Fig. 11. Open area NEU error, number of satellites and GDOP

From the figure can be seen that 10 to 11 satellites are available with a good GDOP value. This good satellite availability and GDOP value results in an RMSe of 0.69 m for the North and 0.67 m for the East direction. A positive error was obtained on the whole trajectory. The vehicle speed was around 50 kph on this trajectory.

The $\overline{RMSe}$ of the 10 experiments is 0.65 m (σ 0.03 m) for the North and 0.68 m (σ 0.08 m) for the East direction.

2) Highway: On a highway we have fewer satellites due to obstructions from trees.

Figure 12 shows the error in the North and East direction, the number of satellites and GDOP value of one of the experiments performed.

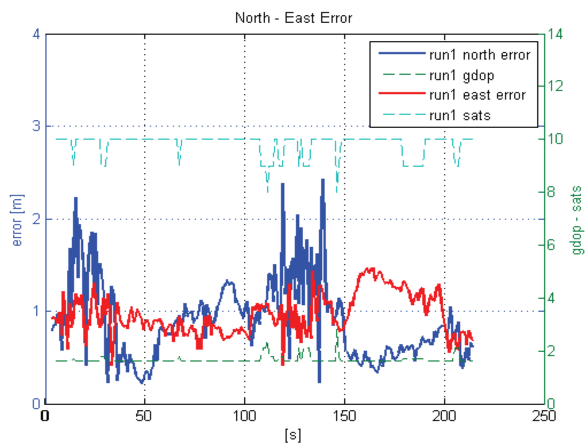

Fig. 12. Highway NEU error, number of satellites and GDOP

From the figure can be seen that not all available satellites could be used, the number of satellites are not constant. Some satellites are obstructed by trees or rejected during the RAIM process. Around the 20th and 130th second a higher variation of the position error can be noticed. At these points in time we had to wait for the traffic lights. A positive error was obtained on the whole trajectory. The maximum allowed vehicle speed on this trajectory was 90 kph. The $\overline{RMSe}$ of the 10 experiments is 1.04 m (σ 0.22 m) for the North and 0.76 m (σ 0.19 m) for the East direction.

Although it is not possible to directly compare the highway against the open area, since the experiment was conducted on a different time, it can be seen that the position is less accurate on the highway than in the open area.

3) Urban: In an urban environment we have fewer satellites available due to obstructions from buildings.

Figure 13 shows the error in the North and East direction, the number of satellites and GDOP value of one of the experiments performed.

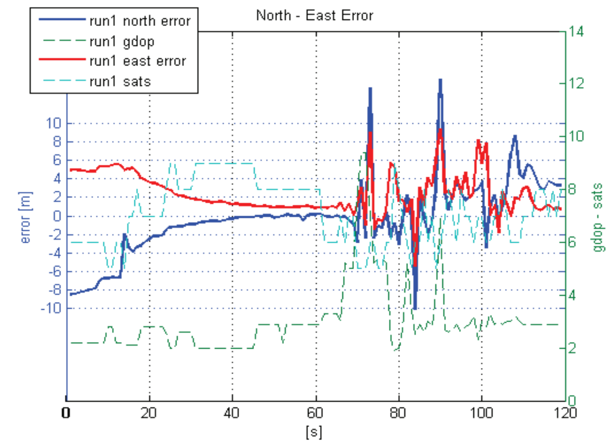

Fig. 13. Urban NEU error, number of satellites and GDOP

The experiments started close to buildings and driving into a more open space. The position error went from more then 5 m to around 1 m. In the open space 9 satellites were received. Hereafter we approached an area with lots of buildings at both sides of the road. A fewer satellites were received which results in a less good position accuracy. The vehicle speed was around 30 kph on this trajectory.

The $\overline{RMSe}$ of the 10 experiments is 3.96 m (σ 0.03 m) for the North and 3.15 m (σ 0.11 m) for the East direction.

The urban environment is far above sub-meter accuracy. Especially for certain regions where many outliers where detected.

B. Influence of the satellite availability

The influence of the satellite availability on the position accuracy is presented. Artificial different combinations are made of satellites and geometries from a real experiment. It makes it easier via this method to compare the different combinations because it will exclude the influence of other dependencies.

The sky plot figure 14 shows the geometry of all available satellites of the experiment. The sky plot, which is a polar plot, shows the azimuth and zenith angles of each satellite.

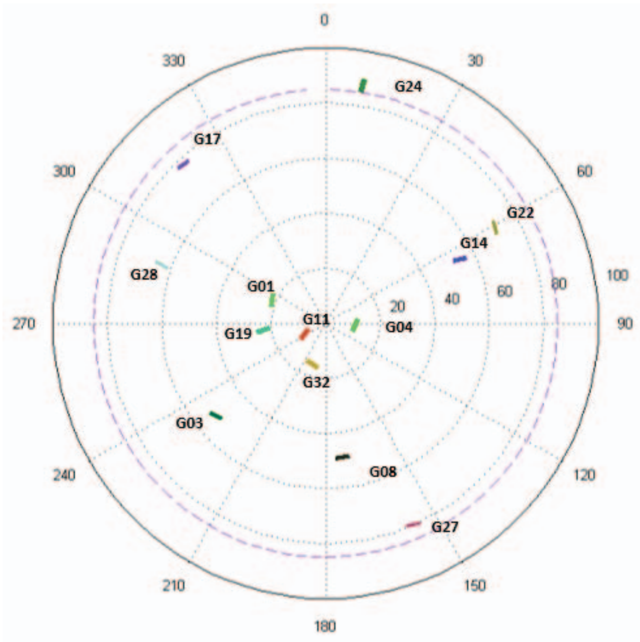

Fig. 14. Available satellites

Table IV shows the computed position accuracy and the $\overline{GDOP}$ for the different combinations. The satellites are grouped from high to low elevation angle.

Number of satellites	North RMSe (m)	East RMSe (m)	$\overline{GDOP}$
5	0.34	0.79	41.6
6	0.59	0.41	11.9
7	0.63	0.39	10.8
8	0.41	0.43	3.6
9	0.26	0.37	2.5
10	0.27	0.37	2.1
11	0.30	0.37	1.7
12	0.20	0.37	1.5

TABLE IV

From the results it can be concluded that the number of satellites will influence the position accuracy. But the most crucial parameter is the geometry of the satellites. Having fewer satellites and good GDOP gives a higher position accuracy than having a lot of satellites with a bad geometry of the satellites.

From some experiments a fixed position offset was noted. This offset has to do with the geometry of the satellites. This could be simulated by only taking for example satellites from the North, a position offset was obtained in the North direction. For satellites only from the East, a position offset was obtained in the East direction.

C. Delays embedded system

How much computation time is consumed by the processing of one epoch and the delay experienced by the client is measured and presented in this topic. It was measured for different environments and tested on different devices. This delay in combination with other delays, like the network latencies, has to be taken into account in the fusion algorithm.

Here is presented the delay from several experiment in different environments on a RPi v2 at 1 Hz (see Fig. 15).

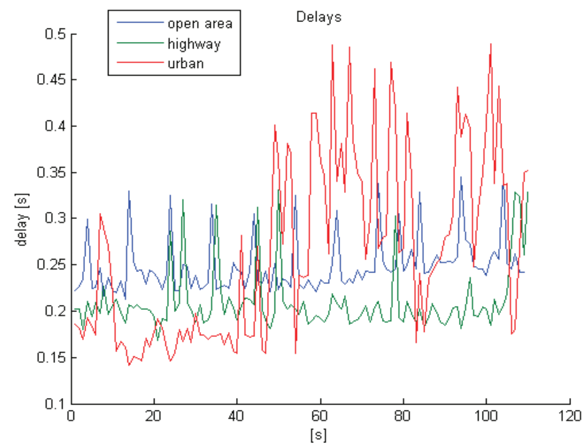

Fig. 15. delays

As we can see there is a $\overline{delay}$ of 250 ms for the highway and open area environment. There are some small spikes in case iterations on a single epoch is required. For the urban environment a much higher delay is obtained due to the RAIM process, the outlier removal.

When deploying the system on different hardware platforms, different delays are obtained (see Fig. 16). It will indicate how much CPU load is taken by the SF-PPP method and what other processes could be deployed on the same platform. It also indicates if it is possible to run at higher samples rates.

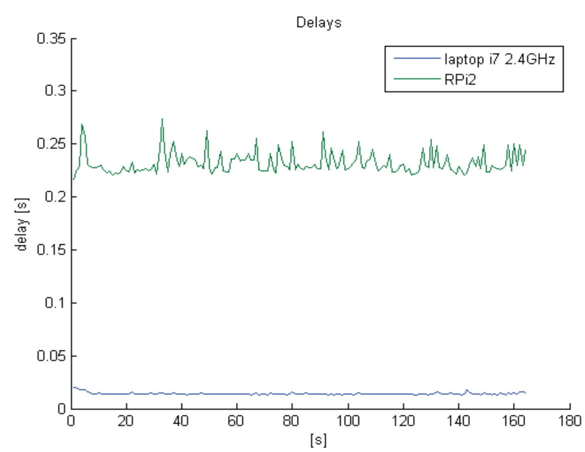

Fig. 16. platforms

On an Intel i7 platform the required sample rate can be achieved. On the RPi v2 a sample rate of 2 Hz can be achieved.

VI. CONCLUSIONS

From the results can be concluded that SF-PPP gives the required sub-meter position accuracy at all times in an open area using a low cost GNSS device. On the highway, sub-meter position accuracy is achieved on about 50 % of the time. In urban environments, which suffers from reflections and poor satellite availability, accuracies varying from 0.5 m up to more than 2 m are achieved.

The SF-PPP solution runs in real-time on an embedded platform (RPi v2) up to 2 measurements per second. The computation time of the position should be improved to provide positions at 10 measurements per second on the same embedded platform.

VII. Future work

Further work is required to improve the position accuracy. Involving a more advanced positioning algorithm: using more satellites from other constellations like GLONASS, using the doppler shift to estimate the vehicle speed, using dual frequency receiver for ionosphere error removal instead of using a predicted corrections and closer integration with other low-cost sensors and a vehicle model. Improving the computation time using hardware acceleration techniques and/or porting to a Digital Signal Processing (DSP) platform in order to achieve 10 measurements per second. Further using a lower cost receiver which only provides raw data without any other features to reduce the total cost. Investigate if a Kalman filter is more accurate and needs less computation power.

Acknowledgment

This work has been supported by the Flanders Make project ’GPS-positioning’, granted by the Institute for the Promotion of Innovation through Science and Technology in Flanders (IWT-Vlaanderen). The provision of the predicted corrections by Andre Hauschild at DLR is acknowledged. The research at the TU Delft was partly funded by ’samenwerkende overheden (ministerie van Infrastructuur en Milieu, provincie Noord-Brabant en het Samenwerkingsverband Regio Eindhoven)’ in the context of ’Beleidsregel 2013 Brabant in-car III’ (the Dynamic Lane Guidance project).

References

[1] *Google maps/streetview*, web, 2015.
[2] Agiv, *Geopunt reference points*, web, 2015.
[3] RJP van Bree and CCJM Tiberius, *Real-time single-frequency precise point positioning: accuracy assessment*, GPS Solutions 16(2 (2012).
[4] P.F. de Bakker, *On user algorithms for gnss precise point positioning. ph.d. thesis, delft university of technology.*, doi:10.4233/uuid:98b1de71-6d51-446a-b161-c8fb58fbcafd, 2016.
[5] P.F. de Bakker, V.L. Knoop, C.C.J.M. Tiberius, and B. van Arem, *Mapping motorway lanes and real-time lane identification with single-frequency precise point positioning*, Proceedings of the Euroean navigation conference (ENC)-GNSS 2014 (Rotterdam, The Netherlands), April 2014.
[6] Bundesamt fuer Geodaesie und Kartographie, *Bnc software, developed under gnu general public license for ’networked transport of rtcm via internet protocol’ (ntrip) to receive gnss streaming data over the internet.*, web, 2016.
[7] Boulaid Boulkroune Norddin El Ghouti Karel Janssen, Erwin Rademakers and Richard Kleihorst, *Bootstrapping computer vision and sensor fusion for absolute and relative vehicle positioning*, Conference on Advanced Concepts for Intelligent Vision Systems, Nov 2015.
[8] AQ Le and CCJM Tiberius, *Single-frequency precise point positioning with optimal filtering*, GPS Solutions 11(1) (2007).
[9] ESA navipedia, *Accuracies of different navigation solutions*, web, 2013.
[10] T. Takasu, *Rtklib: Open source program package for rtk-gps*, web, 2009.