Real-time 2D localization of a car-like mobile robot using dead-reckoning and GPS, with satellite masking prediction

Éric Lucet [*] , David Bétaille [†] , Donnay Fleury Nahimana [†] , Miguel Ortiz [†] , Damien Sallé [*] , Joseph Canou [*]

[*] Robosoft, Technopole d'Izarbel, 64210 Bidart
{eric.lucet,damien.salle,joseph.canou}@robosoft.fr

[†] Laboratoire Central des Ponts et Chaussées, 44340 Bouguenais
{david.betaille,donnay-fleury.nahimana,miguel.ortiz}@lcpc.fr

***Abstract*—The paper deals with the study and the implementation of an extended Kalman filter (EKF) applied to real-time 2D vehicle positioning. This has special applications in fields where inaccurate models or disturbed sensors are involved, such as in driving assistance or mobile robotics in urban area. Particular attention has been payed to the adaptation of the algorithm to the real-time conditions of data acquisition. The contribution of this work also includes some tests performed in real conditions to check the localization algorithm efficiency. To do so, a vehicle is equipped with gyroscope and odometer sensors to be fused with GPS solutions. A Mahalanobis test is also implemented to verify the GPS consistency and reject outliers. Finally, a satellite constellation prediction tool has been used, that encompasses both the planned trajectories and the GPS ephemeris, as well as the surrounding structures.**

Keywords: State Estimation, Extended Kalman Filter, Real Time Implementation, Mobile Robotics, GPS Mask Prediction.

I. Introduction

Many advanced mobile robotics solutions are developed to significantly reduce the cost of services in transport. In that way, the Robosoft company supplies cybercars both as research platforms and as vehicles for touristic sites exploitation (Simserhoff, Vulcania). If the first automatic system for transportation of people were dependent on big infrastructures (rails, buried wires, beacons, etc.), the new ones, thanks to space technologies like GPS, and soon Galileo, require very low infrastructure costs. Robust and accurate localization of the vehicle are therefore required in order to guarantee performance and safety. The CTS-SAT project was developed in this context to expend the field of use of mobile robots for the automatic transportation of people in urban area.

Localization is traditionally carried out using stochastic techniques like the well known Extended Kalman Filter (EKF) that often provides adequate solution to mobile robots localization. There are many localization methods like for example Kacemi et al. (2006), Matia et al. (2004) or Joly (2007), with respect to the process model complexity, the variety of on-board sensors. We purpose here to use RTK DGPS for the absolute localization, encoders and an optical fiber optic single-axis gyroscope (KVH RD2100) for the relative localization, also known as dead-reckoning. The measurement data are fused: these two complementary kinds of sensors enable the mobile robot to localize itself accurately and continuously. We investigate the localization algorithm based on the EKF theory by adapting it to the real-time constraints. In that way, this paper can be linked to the work of Bouvet and Garcia (2000).

A simulation is done in a dynamic environment, using an interaction wheel-soil model of forces designed by Szostak et al. (1988). We will analyze the localization efficiency of the vehicle. This is an electric car designed and manufactured by the Robosoft company which comprises four driven wheels. Each of the four wheels is independently actuated. Wheel sensors, vertical gyroscope turn rate, and GPS solutions are simulated with Gaussian measurement errors.

Then, recorded data of real tests performed in the Cité de l'espace of Toulouse are detailed and filtered with the real-time EKF. These data are captured on-board the vehicle equipped with the sensors described above. These tests made it possible the validation of both the filter studied in simulation and also the prediction of the GPS satellite masking along the trajectory due to buildings around.

This paper is organized as follows. In the second section, the real-time EKF is detailed. In the third section, simulation results using the algorithm are presented and analyzed. In the next section, captured experimental data are described and filtered. The last section focuses on the satellite masking prediction.

II. Real-time algorithm

This section introduces the loosely coupled hybrid GPS / inertial filter designed for the real-time localization of the robot and the information supply of its controller. The EKF algorithm consists of two steps: the prediction by means of the proprioceptive sensors (odometer and gyroscope) at a frequency of 10 Hz; and the use of the RTK DGPS solutions, at 1 Hz if available (i.e. without mask): this is the estimation or correction step. The EKF is a state estimator of a non linear system in a stochastic context. Consequently, an evolution model of the vehicle is needed to predict its position and orientation. A kinematic model of the vehicle is used. In case of GPS masking, the position is obtained from the kinematic model fed with the proprioceptive data, guaranteeing the availability of the system (i.e. the continuous localization).

Real-time introduces some constraints. First is the speed of data processing: it must be done before the next data is received. This is why the code must be optimized. Second constraint is due to GPS solutions that are sent with some latency (around 100 ms) and therefore slightly apply to past position of the vehicle. The typically received NMEA sentences contain the date in which the receiver measurement really took place. In parallel, all other sensors data have to be dated properly. To do so, an acquisition and time-stamping process controlled by the PPS (pulse-per-second) supplied by the GPS receiver has been implemented.

A. Modelisation

The mathematical evolution model is $\dot{X} = f(X,U)$ with X the state vector including the position (x,y) of the GPS antenna and the yaw angle θ in an absolute frame, and with U the input. We use a local frame tangent to the earth, with x toward the east, y toward the north, and the steering angle in the trigonometric direction from the x axis. The kinematic model of the car-like vehicle, at an instantaneous speed denoted V is:

$$\begin{aligned} \dot{x} &= V\cos\theta \\ \dot{y} &= V\sin\theta \end{aligned}$$

After a discretization, $X_{i+1} = f(X_i, U_i)$ the evolution model, is given by :

$$\begin{aligned} x_{i+1} &= x_i + \Delta s_i \cos(\theta_i + \Delta\theta_i/2) \\ &\quad - \Delta\theta_i (T_x \sin\theta_i + T_y \cos\theta_i) \\ y_{i+1} &= y_i + \Delta s_i \sin(\theta_i + \Delta\theta_i/2) \\ &\quad + \Delta\theta_i (T_x \cos\theta_i - T_y \sin\theta_i) \\ \theta_{i+1} &= \theta_i + \Delta\theta_i \end{aligned}$$

T_x et T_y are the axial and transverse offsets between the GPS antenna and the middle of the rear axle.
Δs_i is the curvilinear distance covered during a sampling period T_e.
$\Delta\theta_i$ corresponds to the heading variation computed thanks to the angular velocity measured by the gyroscope, with the equation: $\Delta\theta_i = T_e\omega_i$.
U_i is the input vector of odometry and turn rate:

$$U_i = \begin{pmatrix} \Delta s_i \\ \omega_i \end{pmatrix}$$

The input being asynchronous, the model is calculated based on the gyroscope frequency.

B. Prediction

After linearization at the first order around the point $\left(\hat{X}_{i|i}, U_{mes_i}\right)$, we have:

$$\begin{aligned} X_{i+1} = f(X_i, U_i) &\approx f\left(\hat{X}_{i|i}, U_{mes_i}\right) + F_i\left(X_i - \hat{X}_{i|i}\right) \\ &\quad + J_{ui}(U_i - U_{mes_i}) + V_i \end{aligned}$$

And: $X_{i+1} - \hat{X}_{i+1|i} = F_i\left(X_i - \hat{X}_{i|i}\right) + J_{ui}(U_i - U_{mes_i}) + V_i$

With $F_i = \left.\frac{\partial f(X,U)}{\partial X}\right|_{X=\hat{X}_{i|i}, U=U_i}$;
and $J_{ui} = \left.\frac{\partial f(X,U)}{\partial U}\right|_{X=\hat{X}_{i|i}, U=U_i}$.

V_i is the dynamic noise due to the modelisation errors.
The covariance of the errors of prediction is given by:

$$P_{i+1|i} = F_i P_{i|i} F_i^T + J_{ui} P_{U_i} J_{ui}^T + Q$$

P_{U_i} is the covariance of the input dynamic noises relative to the proprioceptive sensors and Q is the covariance of the modelisation noise. The modelisation errors depend on the vehicle properties and its environment, the phenomena of sliding for example increasing them strongly.

C. Observation

For the observation step, GPS navigation solutions (x_{DGPS}, y_{DGPS}) are directly used to correct the two first components of the state vector. The corresponding equation is thus linear:

$$Y_i = \begin{bmatrix} x_{DGPS_i} \\ y_{DGPS_i} \end{bmatrix} = HX_i + W_i$$

with $H = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix}$ and W_i the noise applicable to GPS solutions, supposed white, gaussian, with a null mathematical expectation and with P_{Y_i} its covariance.

Predictions and observations being asynchronous, a GPS solution is used between time i and $i+1$. The value of angular velocity ω_i is considered constant during the entire $(i, i+1)$ interval.

The time of availability of the navigation solution has been identified as a constraint in the real-time context: this one is not available (i.e. calculated and delivered by the receiver) at the measurement time, but about $100ms$ later. This is why we have to do some predictions until the GPS solution becomes available, apply the correction and recalculate these predictions at the complete availability. A predicted state is computed at the PPS time and is kept in memory to be reused when the next navigation solution has been received. Figure 1 illustrates this process.

D. The Mahalanobis distance criterium

Values of the error models are tunable. The filter tuning mainly concerns the kinematic model noises and the input measurement noises, with the aim of encompassing the real dead-reckoning error with an envelop that corresponds to the standard deviation predicted. When the filter settings are done, a statistic test is performed to detect the erroneous observations

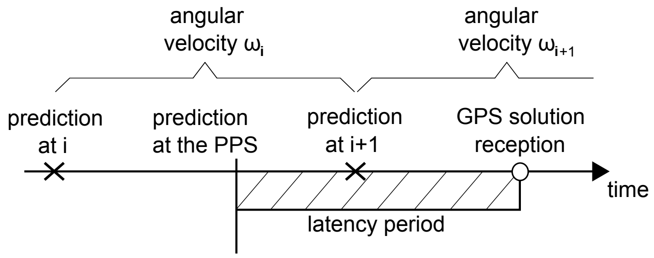


Figure 1. Real-time prediction and estimation chronogram

and eliminate them. It is a test usually based on Normalized Innovation Square (NIS), also called Mahalanobis test.

$$NIS = [Y_i - HXi]^T [HP_{X_i} H^T + P_{Y_i}]^{-1} [Y_i - HXi] < \chi^2$$

The NIS is compared to a threshold given in our case by the two degrees of freedom χ^2 law, for a given $p\%$ level of confidence.

III. Evaluation in simulation

A. *Implementation*

The simulation is executed with RobuBOX, using Microsoft Robotics Developers Studio (MRDS) and Ageia PhysX, a highly realistic 3-dimensional dynamic environment. Robosoft RobuBOX is a software package that allows fast, easy and re-usable development and deployment of robotic applications. It is presented by Sallé et al. (2008) and more detailed by Lucet et al. (2008), Lucet et al. (2009). This software is provided with all Robosoft robots, but can also be used without any hardware platform, as all the RobuBOX software can run indifferently on Robosoft's robotic platforms and in simulation. We select the existing bricks we want to use for hardware management, signal processing, control and application/scenography definition. Then the missing functionalities are developed. The dynamics model of our vehicle (a RobuCAB) is simulated. Gaussian white noise is added onto the sensor measurements according to the simulated hardware. The used values are listed in the table I below.

Table I
Data Table

Description	Value	Unit
Gyroscope offset	0	$^0/s$
Odometer increment	0.0000498	m
Position model standard deviation	0.16	m
Rotation model standard deviation	0.0	^{0}
Gyroscope measure standard deviation	0.0044	$^0/s$
DGPS measure standard deviation	0.4	m
Maximum velocity	10.0	m/s
Minimum velocity	-2.0	m/s
Maximum steer angle	0.5	rad
Minimum steer angle	-0.5	rad
Acceleration	1.0	m/s^2
Deceleration	1.0	m/s^2

The gyroscope measure standard deviation of $0.0044°/s$ results from the $0.083°/\sqrt{h}$ random walk specified by the manufacturer. Also, the $0.16m$ position model standard deviation results from a $0.5m/\sqrt{s}$ random walk, that figures out the possible deviation from the assumption that the robot locally moves circularly without slipping.

B. *Evaluation of the localization algorithm*

The simulation consists of following a curved path on a horizontal ground in an urban area. This path is plotted and a part of it is zoomed in Figure 2. The GPS solutions and the positions computed with the EKF filter previously described are recorded. GPS perturbations as masking phases, multipath effects, atmospheric effects or relativistic effects are not taken into account in this simulator. Only the latency time during the GPS measures delivery is considered.

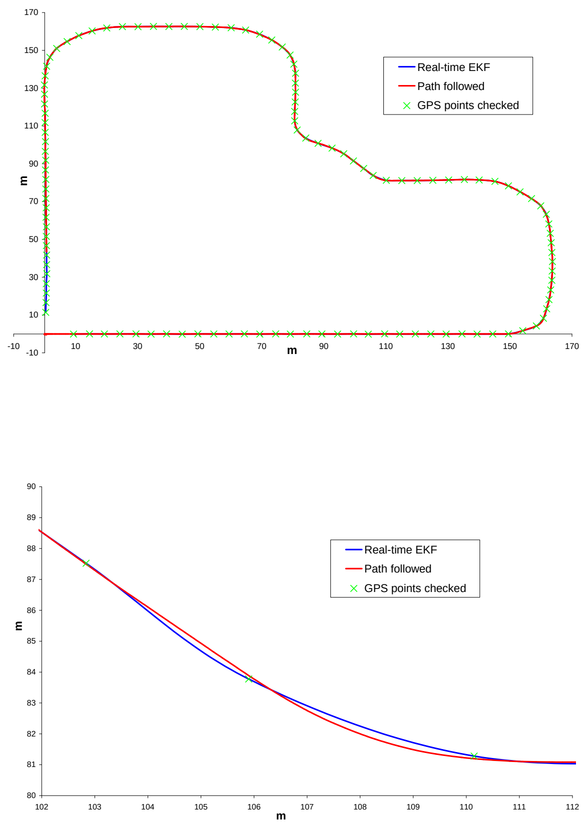


Figure 2. Simulated real-time localization

The path is covered in 116 seconds (approximately 1,93 minutes). The average 99% error is 0.3 meters on both plane axes. The yaw angle being well initialized, good performances are noticeable. The GPS solutions can show a transverse error to the trajectory, but of course also a longitudinal error (if a come back can be seen, it is because of a GPS error toward the rear of the vehicle). After a few time with no GPS solutions, a plane dead-reckoning error is corrected which usually cause an irregularity in the trajectory. This is a perturbation to avoid, especially for a controller implementation. A compromise should be done between the limitation of this perturbation (by smoothing) and the acceptable delay introduced by a smoother (Bétaille (2008)).

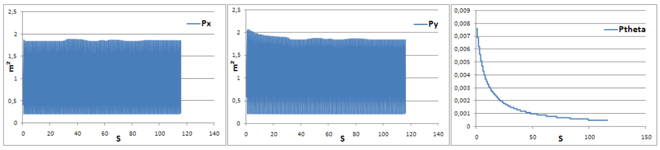


Figure 3. EKF Variances

Figure 3 is the EKF variances recorded during this simulation. The variances of the x position and the y position are oscillating, remaining lower than $2m^2$, and the variance of the yaw is very small, decreasing all over the time.

IV. Experiments

Experiments were performed in the Cité de l'espace of Toulouse in January 2009 with RobuCAB (Figure 4). RobuCAB is an electric car designed and manufactured by the Robosoft society that consists of a four driven wheels with an Ackerman-style steering system on the front wheels. Each one of the four wheels is independently actuated. An acquisition computer embedded the Windows CE operating system. Novatel DLV 3 receiver was used at 1 Hz frequency. The proprioceptive sensors used were a fiber optic gyrometer KVH DSP3000 subsampled at 10 Hz frequency and an encoder on the rear left wheel.

Figure 4. Equipped RobuCAB in the Cité de l'espace

The measurements were EKF filtered. For two masking periods of 13 and 3 seconds shown on Figure 5, 6 and 7, a position error of 0.5 and 0.3 m was recorded. This error may vary depending on the heading error at the beginning of the mask, but once the heading has converged, the drift is rather stable. Since the accuracy specified for roving in the Cité de l'espace is 0.2 m, loosely coupled filter can not deal with GPS masking phases longer than a couple of seconds, a duration when the position error may exceed several decimeters. This is why the experiments have to be planned during good GPS coverage, and by taking into account surrounding possible obstacles.

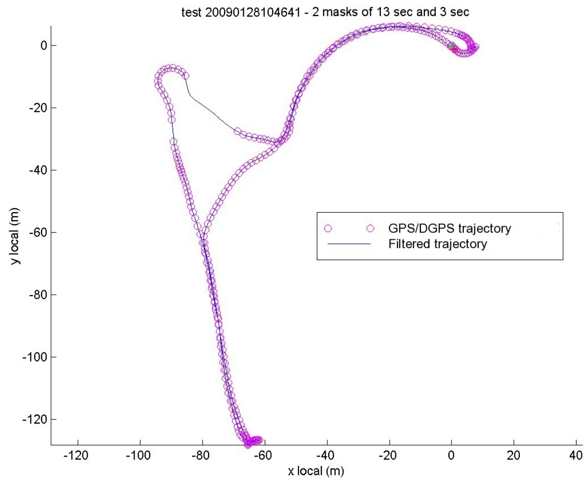


Figure 5. GPS/DGPS and filtered trajectories

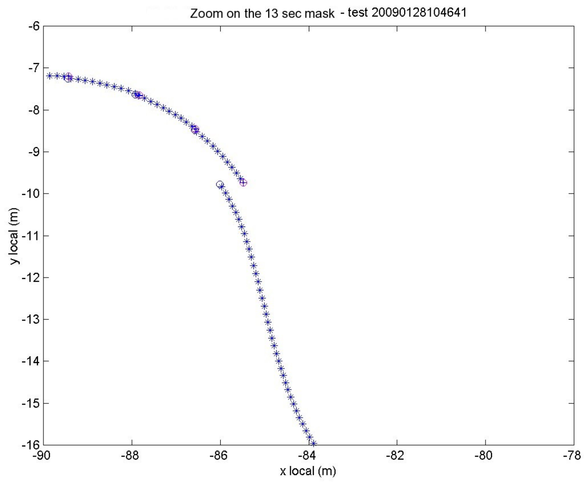


Figure 6. Trajectory: Zoom on masking period of 13 seconds

V. Evaluation of GPS availability

Using a navigation environment 3D model Figure 10, this section focuses on Ergospace simulations in order to evaluate the GPS constellation availability, and, as a consequence, the possible masked periods and locations along the planned trajectories. As the mobile robot moves in masked area during a certain period, the prediction of satellite positions will help to enhance the localization availability. RTK DGPS requires 5 satellites minimum. The Ergospace ray tracing software allows simulation and prediction of the satellite conditions of reception. For each time and on every point of the trajectory, the number of received satellites is computed, considering

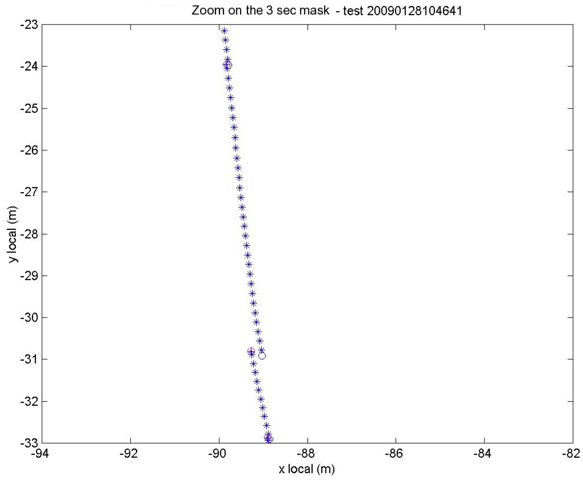
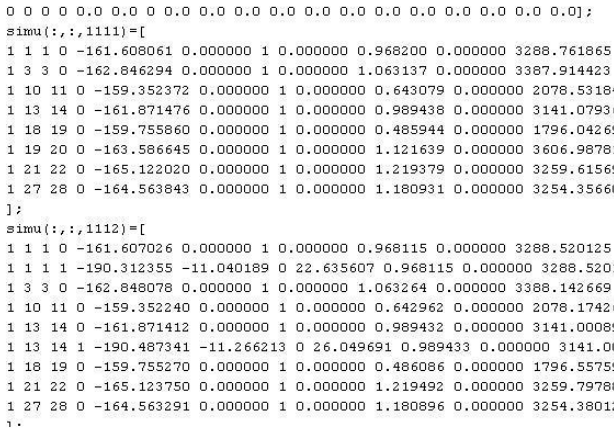


Figure 7. Trajectory: Zoom on masking period of 3 seconds



```
0 0 0 0 0.0 0.0 0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0];
simu(:,:,1111)=[
1 1 1 0 -161.608061 0.000000 1 0.000000 0.968200 0.000000 3288.761865
1 3 3 0 -162.846294 0.000000 1 0.000000 1.063137 0.000000 3387.914423
1 10 11 0 -159.352372 0.000000 1 0.000000 0.643079 0.000000 2078.5318
1 13 14 0 -161.871476 0.000000 1 0.000000 0.989438 0.000000 3141.0793
1 18 19 0 -159.755860 0.000000 1 0.000000 0.485944 0.000000 1796.0426
1 19 20 0 -163.586645 0.000000 1 0.000000 1.121639 0.000000 3606.9878
1 21 22 0 -165.122020 0.000000 1 0.000000 1.219379 0.000000 3259.6156
1 27 28 0 -164.563843 0.000000 1 0.000000 1.180931 0.000000 3254.3566
];
simu(:,:,1112)=[
1 1 1 0 -161.607026 0.000000 1 0.000000 0.968115 0.000000 3288.520125
1 1 1 1 -190.312355 -11.040189 0 22.635607 0.968115 0.000000 3288.520
1 3 3 0 -162.848078 0.000000 1 0.000000 1.063264 0.000000 3388.142669
1 10 11 0 -159.352240 0.000000 1 0.000000 0.642962 0.000000 2078.1742
1 13 14 0 -161.871412 0.000000 1 0.000000 0.989432 0.000000 3141.0008
1 13 14 1 -190.487341 -11.266213 0 26.049691 0.989433 0.000000 3141.0
1 18 19 0 -159.755270 0.000000 1 0.000000 0.486086 0.000000 1796.5575
1 21 22 0 -165.123750 0.000000 1 0.000000 1.219492 0.000000 3259.7978
1 27 28 0 -164.563291 0.000000 1 0.000000 1.180896 0.000000 3254.3801
];
```

Figure 9. Simulation data file: Ergospace file at the epochs when building masked PRN 20 from the vehicle point of view (Transmitter Set ID, Transmitter ID, PRN, reflections, Total attenuation,$\cdots$)

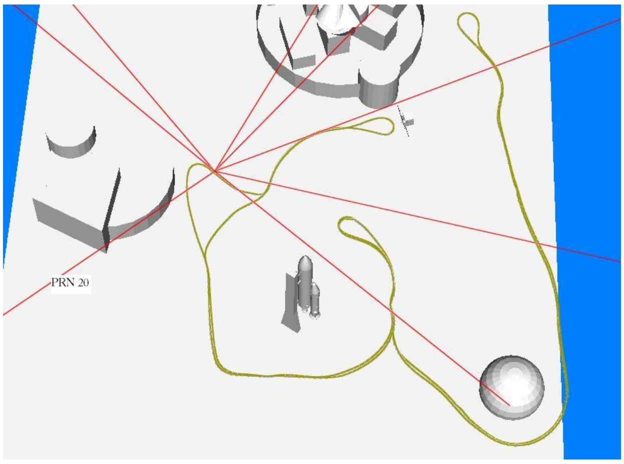
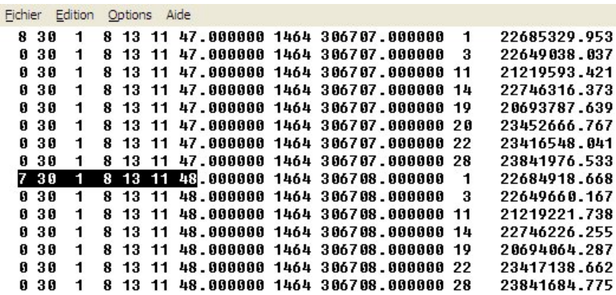


Figure 10. Satellite signals on 3D model; direct rays

here that only direct line-of-sight satellites can be considered. Then the availability of satellite constellation can be calculated according to any start time of planned navigation in a day. Figures 8 and 9 are measurement and simulation data file examples. On the same time as during the measurement conducted on January 30th 2008, the simulation shows the same navigation parameters. As shown on the figures, the PRN 20 is masked by the building at that moment. This has the outcome of reducing the number of satellite used and degrading DOP parameters.



```
Fichier  Edition  Options  Aide
 8 30  1  8 13 11 47.000000 1464 306707.000000  1   22685329.953
 0 30  1  8 13 11 47.000000 1464 306707.000000  3   22649038.037
 0 30  1  8 13 11 47.000000 1464 306707.000000 11   21219593.421
 0 30  1  8 13 11 47.000000 1464 306707.000000 14   22746316.373
 0 30  1  8 13 11 47.000000 1464 306707.000000 19   20693787.639
 0 30  1  8 13 11 47.000000 1464 306707.000000 20   23452666.767
 0 30  1  8 13 11 47.000000 1464 306707.000000 22   23416548.041
 0 30  1  8 13 11 47.000000 1464 306707.000000 28   23841976.533
 7 30  1  8 13 11 48.000000 1464 306708.000000  1   22684918.668
 0 30  1  8 13 11 48.000000 1464 306708.000000  3   22649660.167
 0 30  1  8 13 11 48.000000 1464 306708.000000 11   21219221.738
 0 30  1  8 13 11 48.000000 1464 306708.000000 14   22746226.255
 0 30  1  8 13 11 48.000000 1464 306708.000000 19   20694064.287
 0 30  1  8 13 11 48.000000 1464 306708.000000 22   23417138.662
 0 30  1  8 13 11 48.000000 1464 306708.000000 28   23841684.775
```

Figure 8. Measure data file: Rinex-like raw data file at the epochs when building masked PRN 20 from the vehicle point of view (date, hhmmss, time of week, PRN, C/A, P1, P2, L1, L2 observables)

Figures 10, 11 and 12 show the 3D model during a simulation on Ergospace tool. As the RobuCAB passes close to building, reflections appear (figure 11) and a satellite PRN 20 is masked by the obstacle due to its situation. After a few seconds, as the robot move on, the satellite is received again. On the whole trajectory, this situation is usually encountered according to the obstacle situation and the epoch of simulation.

The section showed the advantage of using simulations on a 3D model of the navigation environment. On a complete day simulation, all places on the mobile track with bad satellite availability come out. The study enables to avoid planning periods when the number of available satellites is not sufficient.

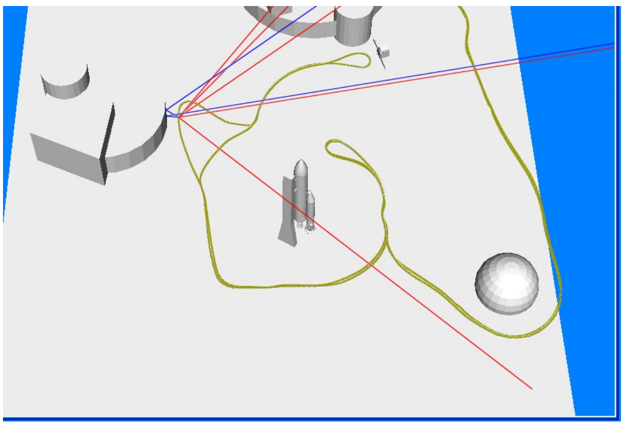


Figure 11. Satellite signals on 3D model close to building; direct rays in red, reflected rays in blue

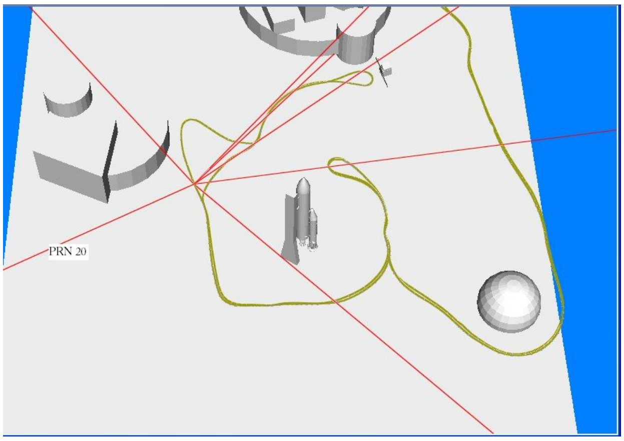


Figure 12. Satellite signals on 3D model. Lack of reflected rays, PRN 20 is masked

VI. Conclusion

In conclusion, a real-time EKF was designed and implemented in simulation and in real conditions. The EKF being a very classical observer, our purpose here has been to adapt it to the real-time conditions by taking into account the RTK DGPS latency. In addition, the input variances are adjusted according to the sensors properties. Several experiments also make it possible an accurate tuning of the modelisation errors too.

Finally, the results of the full scale experiment reported in this article are twofold: first we have checked the algorithm efficiency in real conditions, and second, we have noticed that GPS masks of a few seconds will not be tolerated, because they would cause a few decimeters plane error that exceeds the application requirements.

This last conclusion motivates the use of a sophisticated GPS satellite masking prediction tool, like Ergospace, in the frame of this application.

References

D. Bétaille. Gyrolis : logiciel de localisation de véhicule en posttraitement par couplage gps - gyromètre - odomètre. *Bulletin Spécial Instrumentation des Laboratoires de Ponts et Chaussées*, No. 275, 2008.

D. Bouvet and G. Garcia. Improving the accuracy of dynamic localization systems using RTK GPS by identifying the GPS latency. In *Robotics and Automation, 2000. Proceedings. ICRA '00. IEEE International Conference on*, volume 3, pages 2525–2530 vol.3, 2000.

Cyril Joly. Caractérisation métrologique de capteurs inertiels et fusion avec un récepteur gps egnos en navigation automobile avancée. Master's thesis, Laboratoire Central des Ponts et Chausses, 2007.

Jamila Kacemi, Serge Reboul, and Mohammed Benjelloun. Information fusion in a hybrid tightly coupled gps/dead-reckoning positioning system. *IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems*, art. Heidelberg, Germany, 3-6 September 2006.

E. Lucet, Ch. Grand, D. Sallé, and Ph. Bidaud. Stabilization algorithm for a high speed car-like robot achieving steering maneuver. In *Proceedings of ICRA'08 : IEEE/Int. Conf. on Robotics and Automation*, pages 2540–2545, Pasadena, USA, 19-23 May 2008.

E. Lucet, Ch. Grand, D. Sallé, and Ph. Bidaud. Dynamic velocity and yaw-rate control of the 6wd skid-steering mobile robot roburoc6 using sliding mode technique. In *Proceedings of IROS'09 : the IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, St. Louis, MO, USA, October 2009.

Fernando Matia, Agustin Jimenez, Diego Rodriguez-Losada, and Basil M. Al-Hadithi. A novel fuzzy kalman filter for mobile robots localization. *IPMU*, art. Perugia, Italia, 4-9 July 2004.

Robosoft. www.robosoft.fr.

D. Sallé, E. Lucet, M. Traonmilin, J. Canou, and V. Dupourqué. When space technologies enable services for the autonomous and driverless transportation of persons. In *Toulouse space apply08*, 2008.

H.T. Szostak, W.R. Allen, and T.J. Rosenthal. Analytical modeling of driver response in crash avoidance maneuvering volume ii: An interactive model for driver/vehicle simulation. Technical report, U.S Department of Transportation Report NHTSA DOT HS-807-271, April 1988.