Journal of Engineering Science and Technology Review 11 (3) (2018) 1 - 8

Research Article

JOURNAL OF
Engineering Science and Technology Review

www.jestr.org

Sensor Fusion Tests for an Autonomous Vehicle, using Extended Kalman Filter

Garcia Bedoya O.[1,*] and Ferreira J. V. [2]

[1]Department of Engineering, Universidad de Bogota Jorge Tadeo Lozano, Colombia
[2]Department of Mechanical Engineering, Universidade de Campinas(UNICAMP), Brazil

Received 7 February 2018; Accepted 15 July 2018

Abstract

Intelligent and autonomous vehicle concepts have received a great degree of attention in recent years. In addition to a set of sensors, actuators, mechanism, and components (hardware and/or software), new technologies should be developed in support of Automation, Control, Perception, Localization, and Navigation. This article presents the initial tests with the instrumentation of an autonomous vehicle. The design was implemented making the automation of a Fiat Punto vehicle. The aim of this instrumentation is to increase the safety of the car through a sensor fusion localization using Extended Kalman Filter. This article presents the experimental result of applying this sensor fusion technique over the sensors installed actually on the vehicle. Some conclusions and future improvement over the system are proposed.

Keywords: Autonomous Vehicle, localization, Instrumentation, sensor fusion, Extended Kalman Filter.

1. Introduction

In the last decades, navigation for mobile robots or unmanned vehicles has become a source of countless research contributions. Accident sources, the vehicles are more and more equipped to prevent accidents by the implementation of the new technologies of active safety systems. Therefore, the intelligent vehicles have to act in case of a driver failure. The active safety has the role of accident avoidance. For that, it has to improve the vehicle security conditions by the introduction of the Advanced Driver Assistance Systems (ADAS). We place in this system category the Antilock Braking System (ABS), the Electronic Stability Program (ESP), the speed control, the emergency braking. By default, these systems have to reduce the accident impact based on the vehicle dynamics parameters.

The safety and security of the vehicle are one of the most important issues when designing an autonomous vehicle [1], as well driver assistance systems. Considering this natural evolution from ADAS to autonomous vehicles, the aim of the project is to research the transition phase that consists of a mixture of automated cars with drivers sometimes in the control loop where the vehicle will have in Level 3 of automation [2].

The vehicle called VILMA01 by the initials in Portuguese of Intelligent Vehicle of Laboratory of Autonomous Mobility. This work discusses the preliminaries tests realized with the instrumentation and automation installed on a commercial vehicle. to apply algorithms of sensor fusion to improve the localization of the system. The localization algorithms look for answering the question where the vehicle is? [3].

The paper is presented in five sections. First one is presented briefly the hardware and software architecture, second, the experimental test of the sensors of the vehicle are presented, third, the different sensor fusion techniques used to join together all the sensor and the dynamic model [4] of the vehicle to improve the localization. Finally, some conclusions and futures improvements in the system are presented.

2. System Architecture

The autonomous robotic vehicle, was designed over the concept of layers and functional groups taking account that, this type of architecture facilitates the separation of functions and development, allowing for the definition of interfaces between the various subsystems [5] [6]. Below is described the software architecture and hardware for the development of VILMA01 [7] in response to questions of the layers in the autonomous vehicle.

2.1. Automation Hardware Architecture

The automation of the vehicle was designed to control the steering, the acceleration, the brake and the gearbox of the car (Fiat Punto 2011) over an embedded system. In the steering, the control was made with an electric motor coupled to the steering axis without mechanical modifications, taking account the security of the vehicle. The three others actuators were controlled through electronic circuit and data networks presents the functional blocks of the hardware architecture, where the line arrows represent the five different type of interaction between the blocks: RS232 network (green), electrical signals (blacks) like AI, AO or DIO, wireless network (orange) for the steering torque sensor, CAN network (purple) and Ethernet (Blue) to connect the embedded system to another computer systems. The hardware of the system was defined with interfaces of programming and debug.

[*] E-mail address: olmer.garciab@utadeo.edu.co

ISSN: 1791-2377 © 2018 Eastern Macedonia and Thrace Institute of Technology. All rights reserved.
doi:10.25103/jestr.113.01

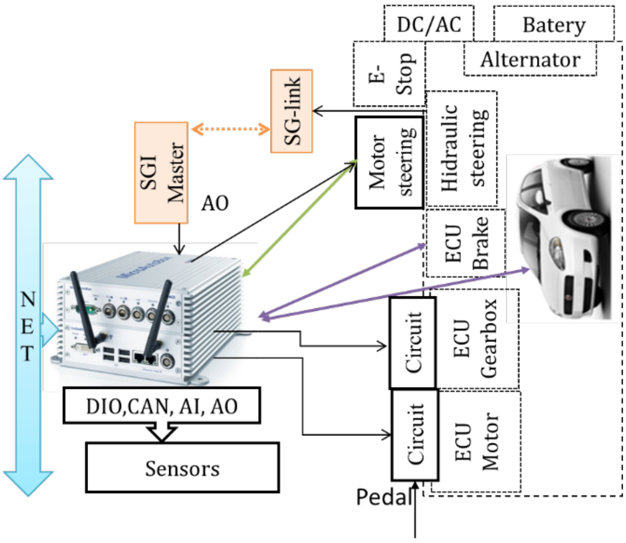


Fig. 1. The hardware architecture of the automation. Communication: Ethernet (blue), RS232 (green), RF (orange), electric signals (black) e CAN (purple).

2.2. Framework architecture

Fig. 2 presents the framework architecture proposed to VILMA01. The framework is divided into three types: firmware, embedded system, and Linux Software according to the hardware programming architecture.

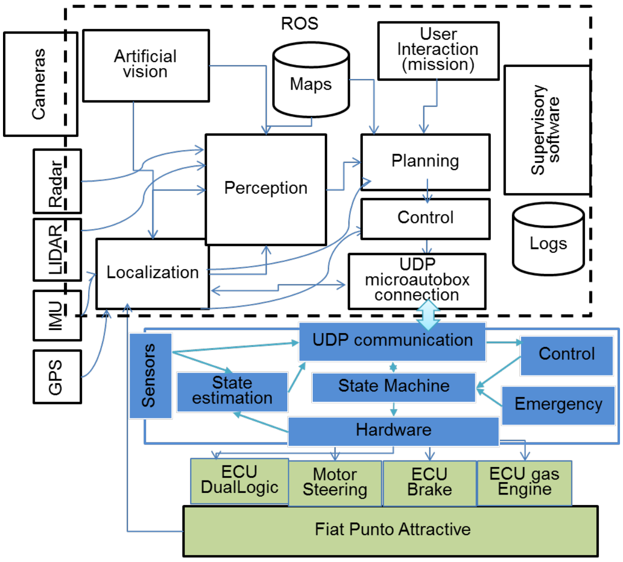


Fig. 2 Framework Architecture of VILMA. Linux software (white), firmware (green), embedded software (blue).

Finally, inside the software architecture, there was integrated a simulator of the vehicle, in the embedded system based on a dynamic model of the vehicle [4] and in the Linux software through a simulator in a 3D environment [8]. The simulators tested on Linux was Gazebo [9], V-Rep [10] and MORSE [11], selecting the last one because open the possibility to work in partnership with the CARINA vehicle [12] from the University of São Paulo - Brazil.

Fig. 3 presents the nodes used by VILMA01 during the experiments that will be presented. Note that the package *vilma_ma_ros* and *vilma_perception* were developed specifically for the vehicle and the others package are available for free in ROS. The package *vilma_ma_ros* implements the nodes required to connect specific hardware to the ROS like the embedded system, which has a designed protocol over UDP to have low-latency and loss tolerating connections. In next section is presented the results of the instrumentation implemented in the vehicle.

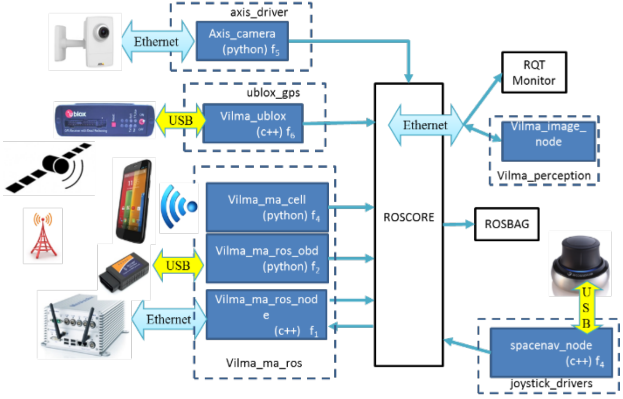


Fig. 3. Nodes executed in ROS for VILMA01

3. Tests in Instrumentation of Vilma01

In order to find the capabilities of the automated vehicle and to determine the feasibility of implementing the cooperative and autonomous techniques in the VILMA01, a series of tests were necessary to analyze the instrumentation data of the vehicle, which is described below. This information was saved with ROS tools and analyzed offline.

3.1. Torque driver sensor

The torque sensors are full Wheatstone Bridge extensometers to measure steel twist without temperature compensation. The sensor is connected to an instrumentation system with included temperature sensor and wireless communication. The sensor was installed behind the steering wheel on the bar as shown in **Fig. 4**.

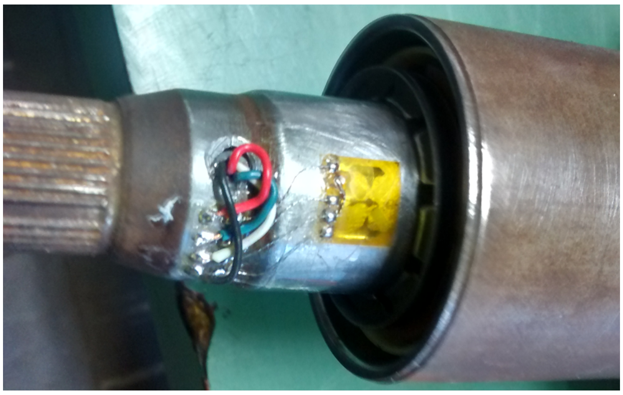


Fig. 4. A sensor mounted on the Steer shaft.

Because the characteristics of the shaft material were unknown, it was not possible to obtain a model of the measured torque, so a calibration of $\mu m/m$ was required for Nm. To obtain resistance variation for$\mu m/m$, The extensometer manufacturer data and the, instrumentation assistant software were used. Thereafter torque data versus $\mu m/m$ were acquired through a digital torque wrench to which a constant load was applied through a mass hung at its end.

The digital torque wrench used in the calibration does not have any communication port, so the measurements were taken visually with a resolution of $\pm 0.1Nm$. The **Fig. 5** presented two results, the first the result measured by the instrumentation, and the second the corrected measure according to the extensometer manufacturer's curves for the experimental temperature of the experiment. As can be seen, the R-square index of the linear regression of the two lines is close to one, so one can approach a line.

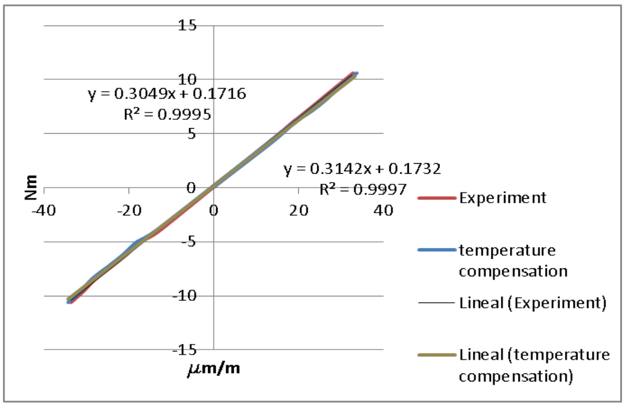


Fig. 5. Curve of Nm vs. µm/m of the torque driver sensor.

The **Fig. 6** describes the way in which the sensor data arrives at the embedded system. In the first instance, the instrumentation amplifies the differential voltage signal, then converts it to a digital signal and transforms its value into $/m$. This value is sent by a wireless signal to a sampling period of 128hz to the receiver that converts the floating point number into volts. This sent value is converted back into MicroAutobox's embedded system with a digital analog converter in a 32-bit floating-point value. Once in the MicroAutobox the value must be converted back to $\mu m/m$ by (1).

$$X = \frac{(5x - 1.5)120\mu m/m}{1.5V} \quad (1)$$

The second step is to perform the temperature compensation, taking into account that the temperature signal has a path similar to the strain gauge signal. Therefore, the corrected amount, call ϵ is described by (2).

$$\epsilon = X + (-15{,}48 + 1{,}72T - 0{,}055T^2 + 0{,}000232T^3) \quad (2)$$

Finally, the value is converted to torque(Nm). Approaching a normal distribution the expected error is approximately$\pm 0.2Nm$. Thus, assuming a confidence interval of 99,6% , equivalent to 3 standard deviations, results in a standard deviation of $\sigma_{\tau_{user}} \cong 0.07Nm$.It should be noted that the Gaussian distribution error assumption is the representation of the following possible errors: aliasing in the signals, temperature, reference torque sensor error, quantization errors, among others.

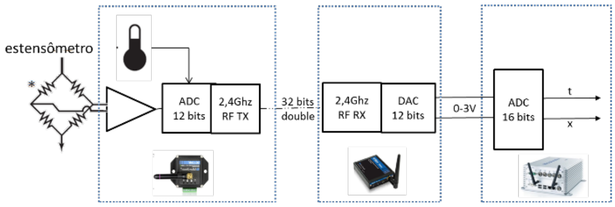


Fig. 6. Torque sensor instrumentation block diagram.

3.2. Accelerometers and barometer sensors

The embedded system has accelerometers on all three axes and a barometer, in addition, the brake ECU that generates the accelerometer information on the vehicle's side and vertical axis. The embedded system was installed on the passenger seat, so the measurements have a difference caused by the angle of inclination of the backrest. The **Fig. 7** shows the values measured during the test path (**Fig. 10**), filtered through a low pass filter (blue) and through the Kalman filter presented in the next section.

From the MicroAutobox sensors, filtered data applying the formulas described in for accelerometers and the barometric formula was obtained altitude at sea level, the angles of rolling and inclination and the altitude of the highway were obtained as shown **Fig. 8**. By way of data verification, the altitude is compared with the GPS value and the angle of inclination is compared with the estimated angle from (3),

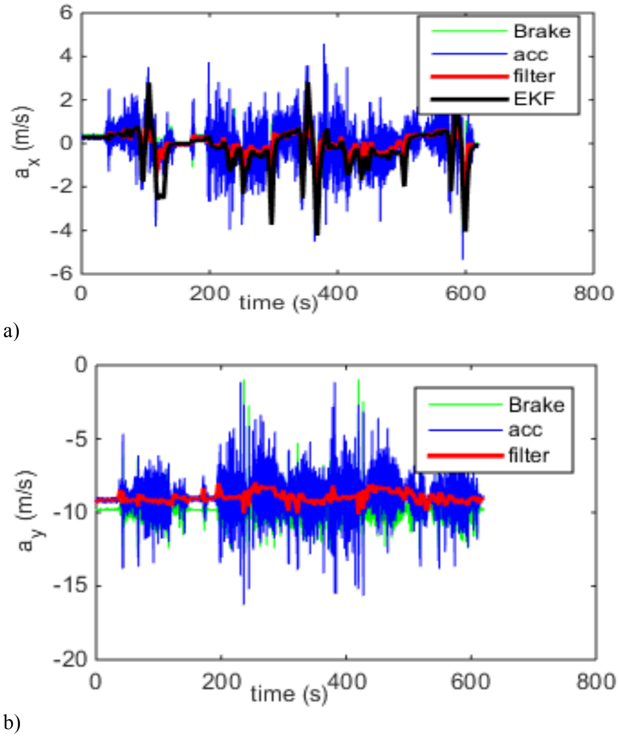
$$pitch_{barometro}(k) \approx \arcsin\left(\frac{h(k) - h(k-1)}{\hat{v}_x(k)Ts}\right) \quad (3)$$

where $h(k)$ is the estimated altitude, $\hat{v}_x$ is the lineal vehicle speed and Ts is the sampling period.

The **Fig. 9** shows the elevation of the road, the rolling angle and the angle of inclination of test path. Note, that the course was performed more than twice and the data are very similar. In the angle of inclination, there are many disturbances being the most relevant those that occur in the spines of the route.

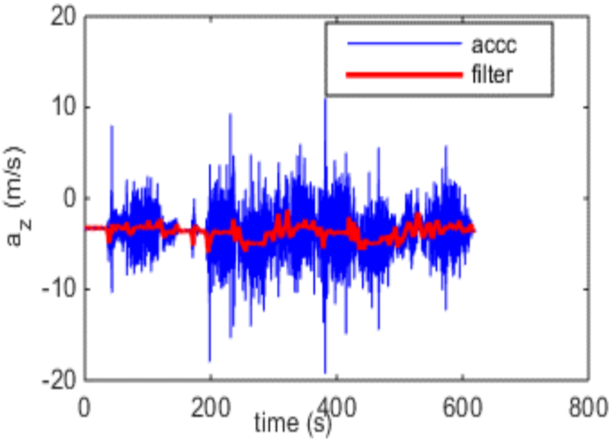
3.3. GPS data

The GPS sensor used was a low-cost EVK-6V reference u-blox designed for vehicles. The **Fig. 10** shows the route made during the test. The values measured by the GPS are compared visually to what is planned by Google Maps through the site. The GPS additionally generates an estimation of the velocity vector, which is used to compare to the speed of the vehicle and from a speed limit is used to estimate the yaw angle of the vehicle. The data of the cell phone connected to the vehicle was not reliable because the Assisted GPS require internet but when the cell phone was connected to the embedded system WiFi network it lost internet and the data was so erratic



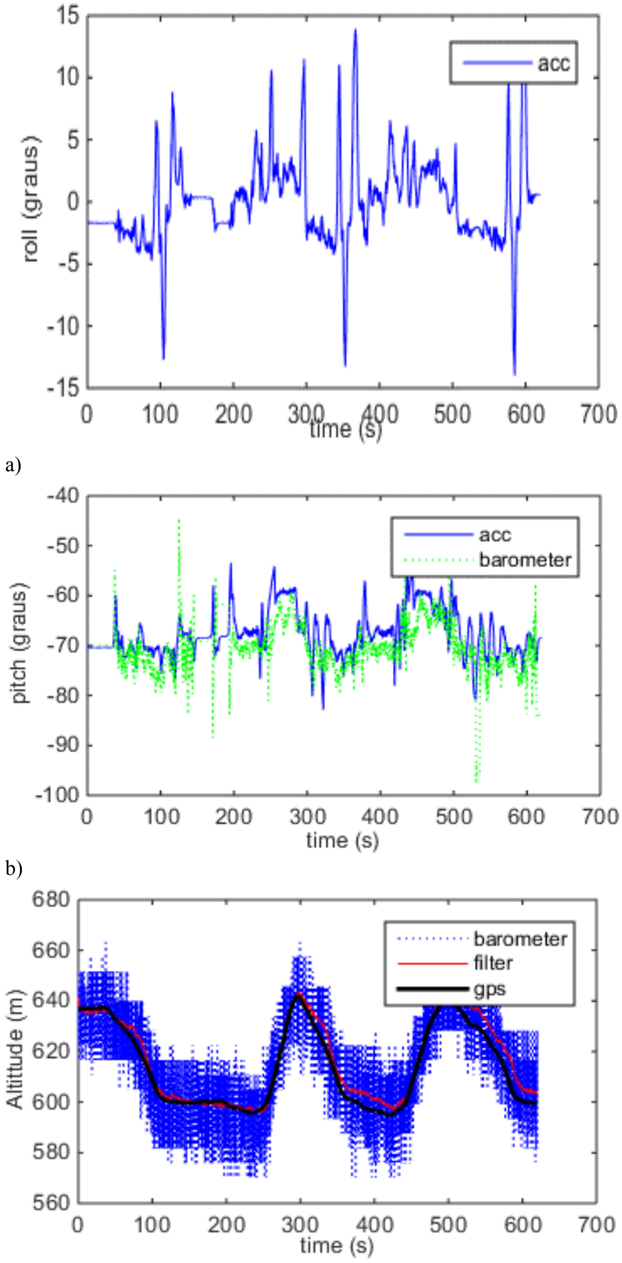
a)

b)



c)

Fig. 7. Accelerometers data.

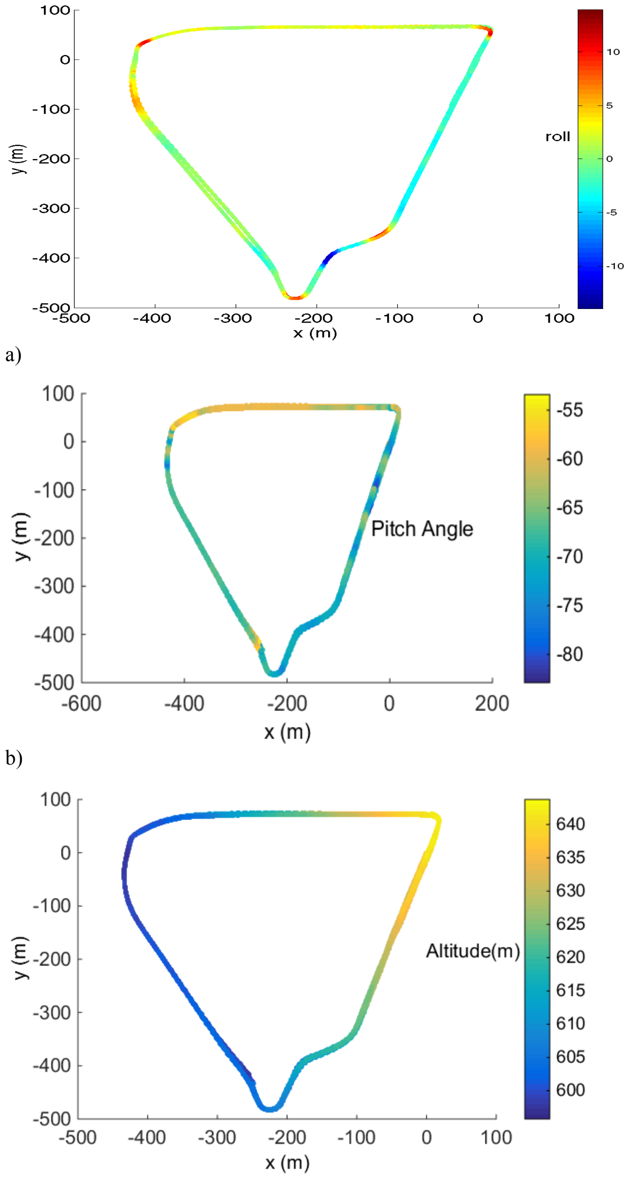


a)

b)

c)

Fig. 8. Altitude, pitch and roll angles estimated during the test.



a)

b)

c)

Fig. 9 Street variables estimated over route one. (a) Roll angle. (b) Pitch Angle. (c) Altitude

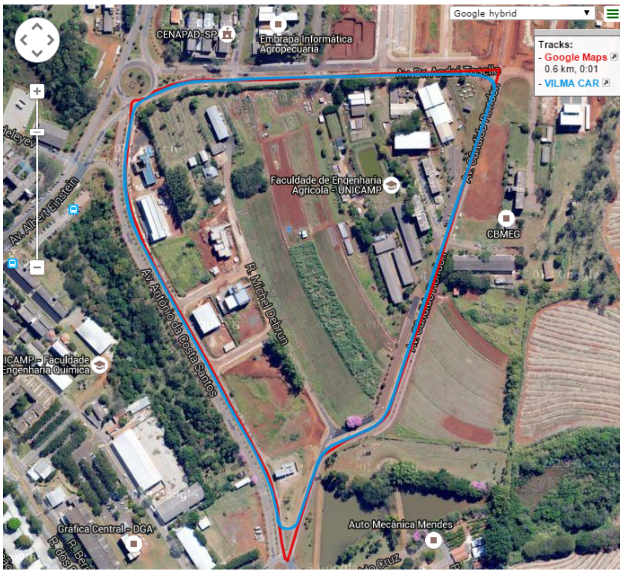


Fig. 10. Comparison between GPS and planned route by Google Maps in test route.

4. Sensor Fusion

4.1. Longitudinal Speed

For the estimation of longitudinal speed, the vehicle has the speed sensors of each of the wheels sent by the brake system through the CAN network, the speed reported by the OBD (on-board protocol) port of the vehicle and the speed estimated by the GPS. In the **Fig. 11** are presented these signals, where it is possible to see that the average of the angular speed of the front wheels (ω_f) multiplied by the theoretical radius of the wheel (reference 195/60 R15 88H) has a ratio of 0.93 with the speed of the vehicle what can be caused by the effective tire radius.

Additionally, it was found that there is a phase shift between the speeds of the wheels and those reported by the GPS and the OBD sensor. This can be explained by the inertia of the vehicle due to the longitudinal dynamics of the vehicle that causes a slip and by the suspension, which generates a response time between the force generated in the iteration of the wheel with the ground and the force perceived by the chassis of the vehicle.

For the estimation of the longitudinal velocity of the vehicle, we used the inverse of (4) [13] through the pseudoinverse of the matrix D.

$$\begin{bmatrix} v_{rl}\\ v_{rr}\\ v_{fl}\\ v_{fr}\end{bmatrix} = \begin{bmatrix} 1 & \frac{-E}{2} & 0\\ 1 & \frac{E}{2} & 0 \\ 1 & \frac{-E}{2}cos(\delta_f)+l_f sin(\delta_f) & sin(\delta_f)\\ 1 & \frac{E}{2}cos(\delta_f)+l_f sin(\delta_f) & sin(\delta_f)\end{bmatrix}\begin{bmatrix} v_x\\ \gamma\\ v_y\end{bmatrix} = D\begin{bmatrix}v_x\\ \gamma\\ v_y\end{bmatrix} \qquad (4)$$

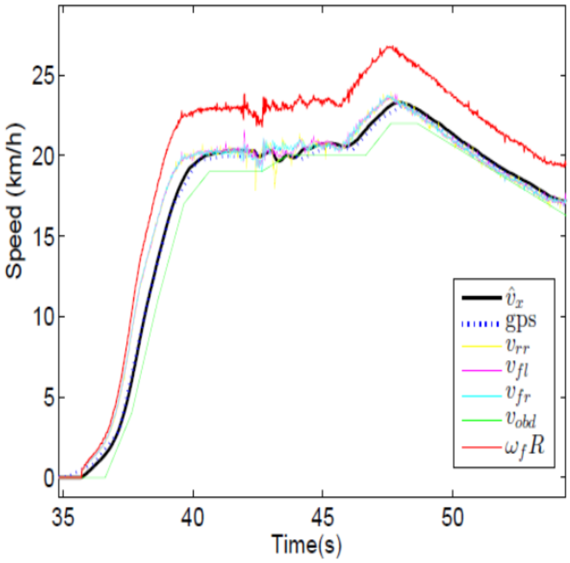
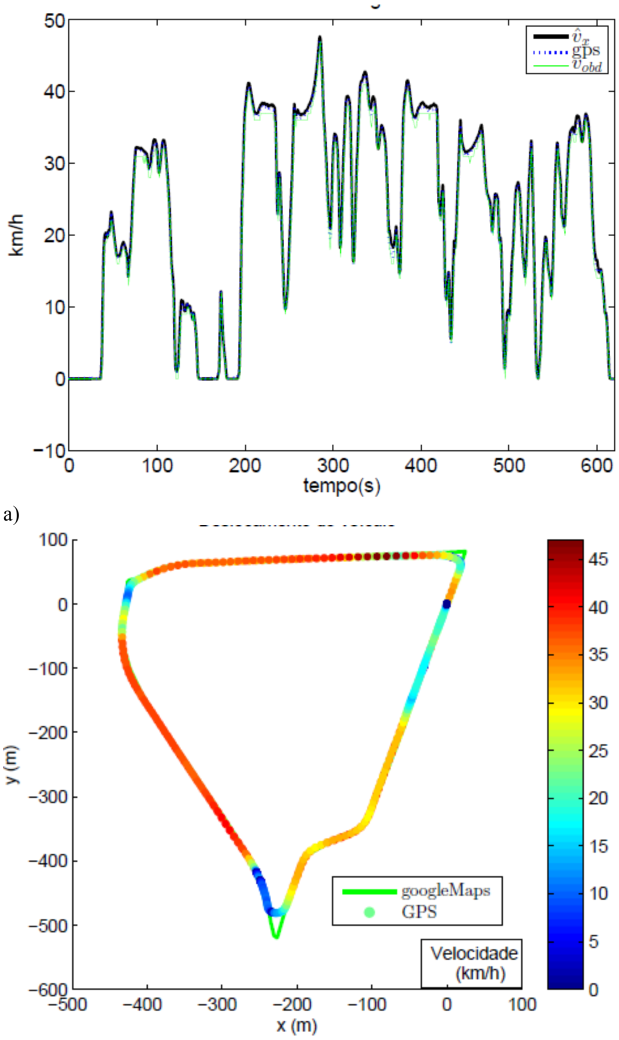


Fig. 11. Speed sensor of VILMA01 in test path.

The estimated longitudinal speed is filtered with a low pass with $\tau = 0.4s$, seeking to model the phase shift. **Fig. 12** shows the longitudinal speed over time and over GPS path. Finally in the **Fig. 13** the yaw speed of the vehicle estimated from the wheel speeds compared to the calculated by the kinematic model.



a)

b)

Fig. 12. Longitudinal speed estimated (a) over time; (b) over path test.

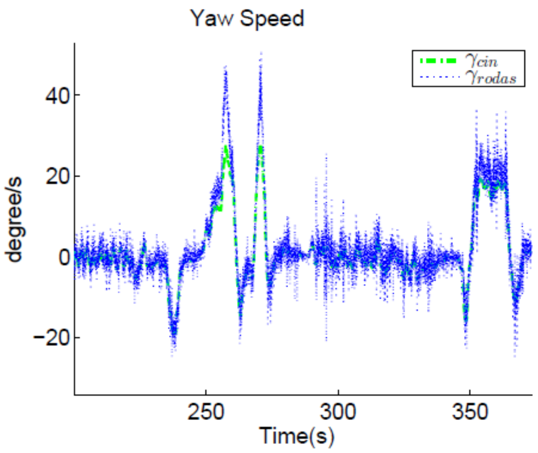
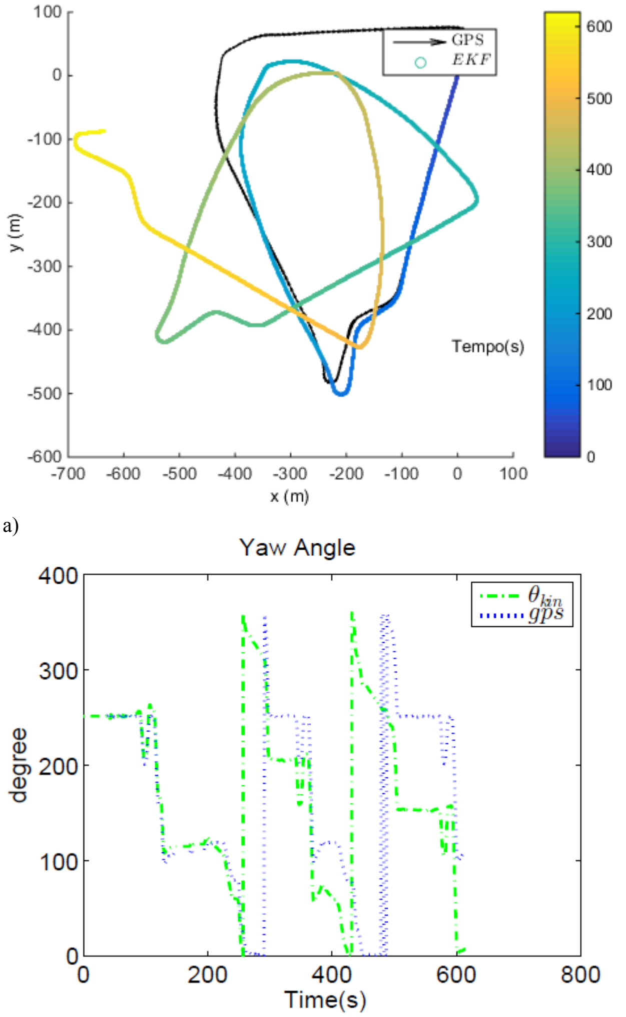


Fig. 13. Angular speed of the vehicle (yaw speed)

4.2. Kinematic Model

From the kinematic model approximation of bicycle model [14], the integration was done by the Runge-Kutta using estimated speed and steering angle. The Fig. 14 shows the results. The errors can be explained because the angle of the steering shaft respect to the front wheel angle has a dead zone around zero that is not being modeled and because in some parts of the course an over current in the steering motor occurred by some tests of cooperative control between the driver and the autonomous systems. Additionally, the dead zone of the gearbox by the twisting generated a slip between the measured angle (of the motor) and the estimated angle (of the steering wheel). The described fault can be observed in the **Fig. 15**. It should be noted that this error has an aggravating factor since the angular speed depends on the longitudinal speed which is approximately $40km/h$ in this area of the course (**Fig. 12**).



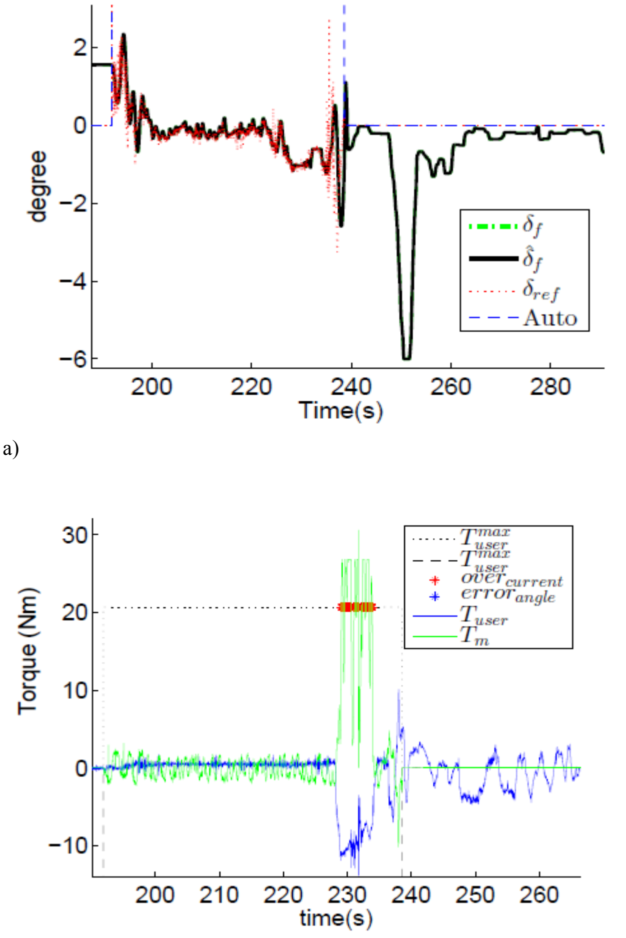
a)

b)

Fig. 14. Wheel angle problem due to cooperative control. (a) Estimated Front Wheel angle. (b) Torques over the steering shaft.

4.3. Extended Kalman Filter for Localization

The Kalman filter can be described as the recursive algorithm to generate an optimal estimate in the statistical sense of the system state variables assuming that the system and the measurement apparatus have a Gaussian noise and that the state model is linear (Siegwart, Nourbakhsh and Scaramuzza 2011). The extended Kalman filter is an approximation for nonlinear models. The state space dynamic model of car described in (Garcia, Ferreira and Neto 2013) (Garcia, Ferreira and Neto 2014), include like state variable the lateral speed, yaw speed, global position, yaw angle and lateral acceleration offset, $\hat{\mathbf{x}}_\mathbf{k} = [v_y \quad \gamma \quad x \quad y \quad \theta \quad a_0]^T$, input vector the steering angle, $\mathbf{u_k} = [\delta_f]^T$ and output vector the sensors,$\mathbf{z} = [a_y + a_o \quad \gamma_{rodas} \quad x_{gps} \quad y_{gps} \quad \theta_{gps}]^T$.



a)

b)

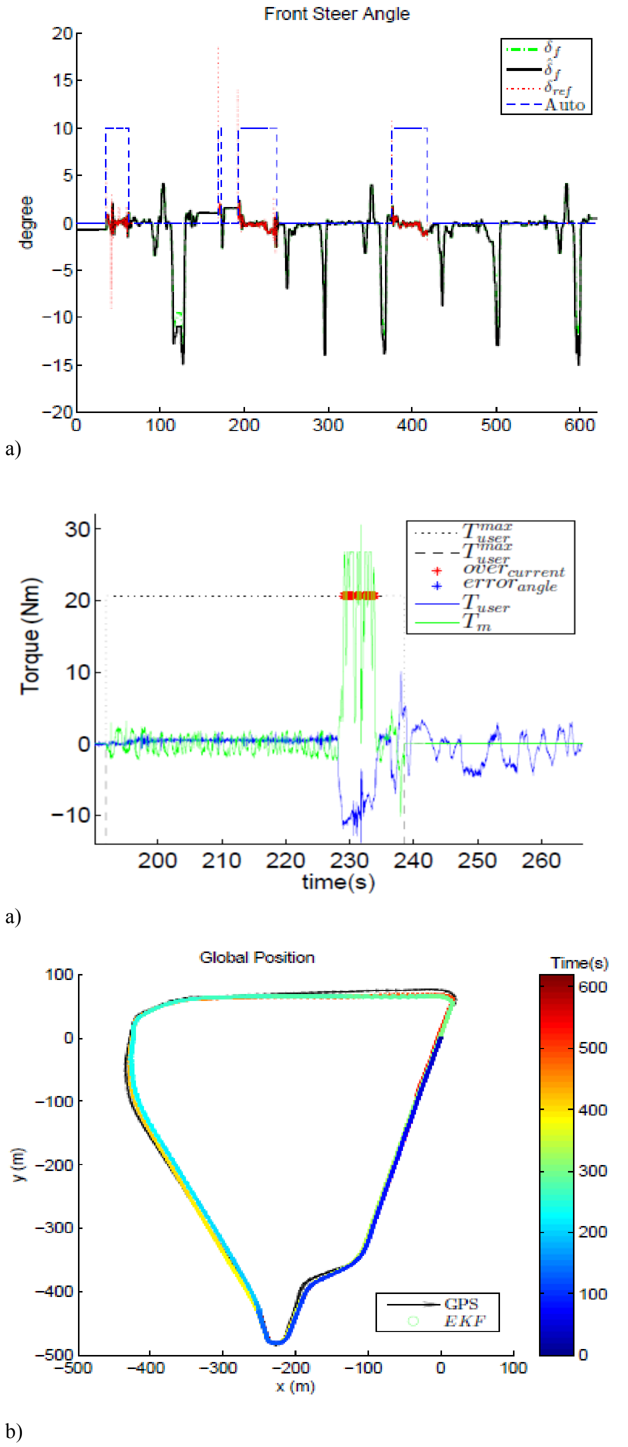
Fig. 15 Wheel angle problem due to cooperative control. (a) Estimated Front Wheel angle. (b) Torques over the steering shaft.

Finally, $\hat{v_x}$ is a parameter in the model. Note that a_0 try to explain the low-frequency data of the lateral acceleration sensor which is caused by the gravity. As parameters was used for the tire model those given in the annex of [16] and the mass and inertia measured of the car do it at Unicamp laboratory. The noise of the model and the sensors are described in equation (5) and (6)

$$\sigma_{vk} = [\sigma_{a_y} \quad \sigma_\gamma \quad \sigma_{x_{gps}} \quad \sigma_{y_{gps}} \quad \sigma_{\theta_{gps}}]^T = [0.2m/s^2 \quad 4°/s \quad 1m \quad 1m \quad 1°]^T \tag{5}$$

$$\sigma_{wk} = [\sigma_{fr} \quad \sigma_{fy} \quad \sigma_{\dot{x}} \quad \sigma_{\dot{y}} \quad \sigma_{\dot{\theta}} \quad \sigma_{\dot{a}_0}]^T = \left[50N\ 50N\ .5\frac{m}{s}\ .5\frac{m}{s}\ .3\frac{rad}{s}\ .3\frac{m}{s^2}\right]^T \tag{6}$$

The **Fig. 16**a presents the front wheel angle (δ_f) estimated from the vehicle's minimum turning radius versus the steering wheel angle is shown assuming a linear relationship and the geometric model of the vehicle [14]. Also shown in this figure are the periods in which the follow-up control of the lines in the path. The result of the estimation of path one is shown in **Fig. 16**b, which presents the same error of the kinematic model around the 220s (cyan color), which is corrected in the second turn (yellow color) due to correction of the state estimator.



a)

a)

b)

Fig. 16. EKF with a dynamic model. (a) Over path test and time (Color) (b) Front wheel angle input to EKF.

It should be noted that the orientation sensor from the GPS speed and the lateral acceleration sensor are only used when the vehicle speed is greater than $1.5 m/s$. One problem found was that when the vehicle turns to the right sub estimates the yaw speed, but when it turns to the left there is an over estimation of the yaw speed. This can be explained by an error in the parameters of the wheel forces or the necessity to include roll and pitch angles in the model. Trying to solve the problem of the forces was used the formula of Ackerman [17] to calculate the lateral force of each front wheel, but this did not solve the problem.

5. Conclusion

From the data analysis of the current instrumentation of VILMA01, it was found that it is possible to do the integration of sensors, which significantly improves the performance of each of the measurements performed. For example, with the merging of longitudinal speed sensors, it is possible to obtain the angular speed of the vehicle. Additionally, it was possible to increase the sampling frequency of the global position from the sampling frequency of the GPS ($1\ Hz$) to the sampling frequency of the proprioceptive sensors ($100\ Hz$). This was obtained taking account the dynamic model of the vehicle.

In vehicle instrumentation, it is necessary to include a vehicle steering angle sensor, as well as to have a longitudinal dynamics model. To improve the estimation of the angles of rolling, yaw, and tilt of the vehicle it is recommended to have an inertial measurement unit that allows obtaining angular speed in all axes. The roll and tilt angles should be included in the dynamics model of car to improve global position estimation through the acceleration vector $vx\omega$. Finally, at present, the transformation of latitude / longitude in meters of the GPS measurement is done by the global reference model system [18] with the ROS Geodesy package, but it is necessary to revise the parameters of the same to be adapted to the Brazilian location.

Acknowledgements

The authors would like to acknowledge the support granted by CAPES and CNPq - processes PDSE:9129/12-0 and SWE:209656/2013-1. O. Garcia was Ph.D. Scholarship PEC/PG CAPES/CNPq-Brazil during part of the project.

References

[1] J.-U. Park, B.-H. Bae, J.-W. Lee and J.-H. Kim, "Design of failsafe architecture for unmanned ground vehicle," in Control Automation and Systems (ICCAS), 2010 International Conference on, 2010.
[2] NHTSA, "US Department of Transportation Releases Policy on Automated Vehicle Development," 2013.
[3] R. Siegwart, I. R. Nourbakhsh and D. Scaramuzza, Introduction to autonomous mobile robots, second ed., MIT press, 2011.
[4] O. Garcia, J. V. Ferreira and A. M. Neto, "Dynamic Model Of A Commercial Vehicle For Steering Control And State Estimation," in XI Simposio Brasileiro de Automação Inteligente (SBAI), 2013.
[5] Y.-L. Chen, V. Sundareswaran, C. Anderson, A. Broggi, P. Grisleri, P. P. Porta, P. Zani and J. Beck, "TerraMax: Team Oshkosh urban robot," Journal of Field Robotics, vol. 25, pp. 841-860, 2008.
[6] S. Thrun, M. Montemerlo, H. Dahlkamp, D. Stavens, A. Aron, J. Diebel, P. Fong, J. Gale, M. Halpenny, G. Hoffmann and others, "Stanley: The robot that won the DARPA Grand Challenge," Journal of field Robotics, vol. 23, pp. 661-692, 2006.
[7] O. Garcia, G. B. Vitor, J. V. Ferreira, P. S. Meirelles and A. de Miranda Neto, "The VILMA intelligent vehicle: an architectural design for cooperative control between driver and automated system," Journal of Modern Transportation, Apr 2018.
[8] T. Ferreira, O. Garcia and J. Vaqueiro, "Software Architecture For An Autonomous Car Simulation Using Ros, Morse & A Qt Based

Software For Control And Monitoring," in XII Simpósio brasileiro de automação Inteligente, 2015.

[9] N. Koenig and A. Howard, "Design and use paradigms for gazebo, an open-source multi-robot simulator," in Intelligent Robots and Systems, 2004.(IROS 2004). Proceedings. 2004 IEEE/RSJ International Conference on, 2004.

[10] M. Freese, S. Singh, F. Ozaki and N. Matsuhira, "Virtual robot experimentation platform v-rep: a versatile 3d robot simulator," in Simulation, modeling, and programming for autonomous robots, Springer, 2010, pp. 51-62.

[11] G. Echeverria, N. Lassabe, A. Degroote and S. Lemaignan, "Modular open robots simulation engine: Morse," in Robotics and Automation (ICRA), 2011 IEEE International Conference on, 2011.

[12] L. C. Fernandes, J. R. Souza, G. Pessin, P. Y. Shinzato, D. Sales, C. Mendes, M. Prado, R. Klaser, A. C. Magalhães, A. Hata and others, "CaRINA intelligent robotic car: Architectural design and applications," Journal of Systems Architecture, vol. 60, pp. 372-392, 2014.

[13] M. Doumiati, "Estimation embarquée des efforts verticaux et latéraux dún véhicule pour le diagnostic du comportement sur route," 2009.

[14] J. M. Snider, "Automatic Steering Methods for Autonomous Automobile Path Tracking," 2009.

[15] O. Garcia, J. V. Ferreira and A. M. Neto, "Design and Simulation for Path Tracking Control of a Commercial Vehicle Using MPC," in Robotics: SBR-LARS Robotics Symposium and Robocontrol (SBR LARS Robocontrol), 2014 Joint Conference on, 2014.

[16] P. Yih, "Steer-By-Wire: Implications For Vehicle Handling And Safety," USA, 2005.

[17] R. A. Cordeiro, J. R. Azinheira, C. de Paiva and Ely and S. S. Bueno, "Efeitos da Dinâmica Tridimensional no Controle de Trajetória de Um Ve\́iculo Robótico Terrestre com Quatro Rodas," in XIX Congresso Brasileiro de Automática, CBA, Campina, 2012.

[18] R. a. M. F. Ruiz, "Desenvolvimento de um sistema de localização híbrido para navegação autônoma de veículos terrestres em ambiente simulado," Campinas, 2014.