

Article

Vehicle Localization Kalman Filtering for Traffic Light Advisor Application in Urban Scenarios

Daniele Vignarca , Stefano Arrigoni  and Edoardo Sabbioni * 

Department of Mechanical Engineering, Politecnico di Milano, 20156 Milan, Italy; daniele.vignarca@polimi.it (D.V.); stefano.arrigoni@polimi.it (S.A.)

* Correspondence: edoardo.sabbioni@polimi.it

Abstract: The recent advancements in Intelligent Transportation Systems (ITS) have revealed significant potential for enhancing traffic management through Advanced Driver Assist Systems (ADASs), with benefits for both safety and environment. This research paper proposes a vehicle localization technique based on Kalman filtering, as accurate positioning of the ego-vehicle is essential for the proper functioning of the Traffic Light Advisor (TLA) system. The aim of the TLA is to calculate the most suitable speed to safely reach and pass the first traffic light in front of the vehicle and subsequently keep that velocity constant to overcome the following traffic light, thus allowing safer and more efficient driving practices, thereby reducing safety risks, and minimizing energy consumption. To overcome Global Positioning Systems (GPS) limitations encountered in urban scenarios, a multi-rate sensor fusion approach based on the Kalman filter with map matching and a simple kinematic one-dimensional model is proposed. The experimental results demonstrate an estimation error below 0.5 m on urban roads with GPS signal loss areas, making it suitable for TLA application. The experimental validation of the Traffic Light Advisor system confirmed the expected benefits with a 40% decrease in energy consumption compared to unassisted driving.

Keywords: vehicle localization; Kalman filter; ADAS; kinematic model; GPS; TLA; ITS



Citation: Vignarca, D.; Arrigoni, S.; Sabbioni, E. Vehicle Localization Kalman Filtering for Traffic Light Advisor Application in Urban Scenarios. *Sensors* **2023**, *23*, 6888. <https://doi.org/10.3390/s23156888>

Academic Editors: Yafei Wang, Chao Huang, Hailong Huang, Zhiqiang Zuo, Bo Leng and Peng Hang

Received: 30 June 2023

Revised: 31 July 2023

Accepted: 1 August 2023

Published: 3 August 2023



Copyright: © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Vehicle localization represents a fundamental task in many fields, ranging from Autonomous Vehicles (AV) to Advanced Driving Assistance Systems (ADASs) as well as traffic management [1]. Indeed, the starting point for most of the control logic, on both the vehicle and infrastructure sides, is the knowledge of the vehicle position. This is typically fed to algorithms intended to compute either vehicle optimal trajectory and speed or safety risk indexes to safely overcome dangerous situations, especially in urban scenarios.

Vehicle localization techniques can be distinguished between the onboard sensor-based systems and those relying upon Vehicle-to-Vehicle (V2V) and Vehicle-to-Infrastructure (V2I) communication. The former category can be further split into active sensor based (e.g., LiDAR and RADAR) and passive sensor based (e.g., GPS and IMU). On the one hand, active sensors are generally more costly and computationally expensive than passive ones. On the other hand, Inertial Measurement Units (IMUs) suffer from noisy signals which may lead to integration divergence, while when dealing with Global Positioning Systems (GPS) a typical issue is the signal loss [2]. In vehicle localization, GPS outage is thus a relevant phenomenon to cope with, especially in urban scenarios, where the presence of trees and high buildings limit the sensor capabilities.

The Green Light Optimal Speed Advisory (GLOSA) system is an application that conveys speed references to the driver to achieve lower travel times, fuel/energy consumption, and safer travel conditions [3]. This can be achieved thanks to the knowledge of road data, the vehicle state in terms of position and speed, and traffic light schedules. The speed profile calculation is typically addressed by taking into account different criteria.

One typical approach is to minimize the engine power demand and idling time [4]. Another approach prioritizes driver annoyance reduction by minimizing the difference between the suggested and actual speeds, or by aiming to pass the traffic light in the smallest amount of time [5].

In this framework, the present study proposes a multi-sensor multi-rate vehicle localization technique based on Kalman filtering as the ego-vehicle position is a prerequisite for making ADASs work. The aim is to have an accurate state estimation that provides the vehicle position to the Traffic Light Advisor (TLA) system thought to be used in challenging scenarios such as urban roads, where pure GPS information may be neither present nor reliable. The field tests for the localization algorithm demonstrated good accuracy results in different conditions and GPS signal availability, proving to be consistent for running ADASs, such as the TLA. Furthermore, having implemented the localization algorithm and knowing the working plans of a set of traffic lights on a predefined path of the city of Milan, the paper reports full-scale testing on a trolley-bus in the urban scenario of the TLA developed by [6] at Politecnico di Milano, confirming the previously obtained simulation results.

The remainder of this paper is structured as follows: After the literature review in Section 2, the experimental setup used for the experimental campaigns is presented in Section 3. Section 4 details the proposed localization algorithm, and Section 5 summarizes the main feature of the TLA and presents the scenarios and the metrics adopted for the ADAS validation. The results of both the localization algorithm and the TLA system are reported in Section 6 while Section 7 draws the conclusions of this work, proposing some future development for the implemented systems.

2. State of the Art

In the literature, the criteria employed for the determination of the speed profile in GLOSA applications typically involve the minimization of the total energy consumption and travel time [7–10]. This kind of ADAS can be further distinguished into two categories based on the number of traffic lights they consider in real time to determine the recommended speeds: single-segment GLOSA (S-GLOSA) and multiple-segment GLOSA (M-GLOSA). S-GLOSA systems focus on analyzing the first traffic light encountered by the vehicle, while M-GLOSA systems consider multiple traffic lights along the vehicle's route. In the case of S-GLOSA algorithms, they typically employ modeling approaches incorporating velocity profiles both upstream and downstream of the intersection, as demonstrated in [11]. However, in recent years, data-driven approaches have emerged as a promising alternative, as highlighted in [12]. In that research, a conventional S-GLOSA system is contrasted with reinforcement learning (RL) implementation, which incorporates data from a single traffic light and limited information from the preceding three vehicles. The RL-based approach resulted in an 11% increase in energy savings compared to the standard S-GLOSA system. These developments in data-driven methodologies highlight the potential to enhance the performance and energy efficiency of GLOSA systems.

In the future, Vehicle-to-Infrastructure (V2I) communication will be one of the main drivers making this kind of Advanced Driving Assistance System possible by providing a great amount of data about adjacent vehicles and traffic [13]. In fact, the work in [14] analyzes the effect of the GLOSA system running different simulations varying both infrastructure variables (e.g., cycle times and communication range) and external variables, such as traffic conditions. As far as traffic is concerned, the research conducted in [15] demonstrates the effectiveness of M-GLOSA systems compared to single-segment approaches, especially in free-flow traffic conditions. However, optimizing M-GLOSA systems while considering traffic light phase changes presents challenges, leading to non-convex feasible solution domains. To tackle this issue, the literature proposes various approaches. Studies like [8,15] have implemented Genetic Algorithms (GAs) to address the optimization problem. Additionally, search-based algorithms, employing semi-heuristic or brute-force methods, have also been explored in [11]. A widely adopted alternative is Model Predic-

tive Control (MPC), and the authors in [16] investigated the MPC application for GLOSA implementation in road segments containing multiple traffic lights.

Self-localization has been extensively studied in the literature for a long time, as it serves as a crucial component in the development of Cooperative Active Safety Systems and ADASs in general. In a comprehensive review conducted in [17], various sensor-based and communication-based approaches for localization are thoroughly examined, with a specific focus on accuracy and real-time performance. The findings of the survey show that data fusion techniques, such as the integration of onboard passive sensors and Vehicle-to-Everything (V2X) communication, offer a promising solution due to their robustness, accuracy, and ability to operate in real time.

GPS outages represent a huge limiting point in vehicle localization in urban canyoning; thus, the most adopted strategy to cope with this issue is to use the Kalman filter and its variants to estimate the vehicle state also when GPS is not available, fusing different sensors (e.g., Inertial Measurement Unit and Wheel Speed Sensor) and a dynamic vehicle model [18,19]. In [2], an extended Kalman filter (EKF) fusing a digital map, IMUs, GPS data, and cellular Base Transceiver Stations (BTS) signals is presented. In their work, the authors proposed the use of cellular BTS not only to overcome GPS outages but also to improve localization accuracy.

Recently, LiDAR and vision sensors have been adopted to overcome the challenges of localization in urban scenarios, fusing these types of sensors to cope with the limitation of every single device [20]. In [21], a cascading Kalman filter and dynamic object removal model using multi-GNSS, INS, Precise Point Positioning (PPP), and vision to improve vehicle navigation performances in urban scenarios is presented. In this framework, a novel application in state estimation is Simultaneous Localization and Mapping (SLAM) which can be involved either in AV applications or in Cooperative ITS. On the one hand, Bersani et al. [22] presents an integrated system for vehicle state estimation using unscented Kalman filter fusing data from different passive sensors, such as GPS and IMUs, and from active sensors, like RADAR and LiDAR, which are used to detect and track obstacles as well as improve the localization algorithm. On the other hand, Wang et al. [23] compares different localization systems based on both GNSS and V2X communication for inter-vehicle distance calculation, which is needed for safety ADAS applications.

An interesting challenge in fusing different sensors is dealing with their different sampling frequency. In fact, in asynchronous multi-sensor systems, there is the possibility to miss some data when performing state estimation [24]. The authors in [25] presented different multi-rate multi-sensor models for Kalman filtering with missing measurements. The idea is to run the state estimation algorithm at the fastest sensor frequency and just predict the state vector whenever a sensor is missing or it is considered not reliable.

In addition to the widely studied Kalman filtering techniques, which are extensively covered in the vehicle localization literature, alternative strategies like graph optimization have also been explored. The authors in [26] introduced a novel approach in their work, presenting a multi-sensor fusion method formulated as a graphical model. This model optimizes the integration of onboard sensors to enhance positioning performance, utilizing a kinematic vehicle model as the underlying basis.

Recently, the advent of the 5G network cooperative oriented the research toward new horizons, such as cooperative localization and the Internet of Vehicles (IoV). In fact, thanks to the perception algorithm of the surrounding vehicles, vehicle communication allows gathering localization information that can be used to either improve the ego-vehicle state estimation when GPS is not accurate [27] or to fill the gap in case the GPS signal is missing for long periods [28]. Another approach investigated in the literature in the last decades is the use of Assisted-GPS (A-GPS) systems which exploit the terrestrial communication link to determine the current location [29]. This type of localization is typically employed on cell phones to avoid decoding the GPS messages for each satellite observed, thus using a remote server [30]. More recently, an alternative A-GPS system combining a barometer and accelerometer is proposed in [31] to improve the localization on smartphones.

This research paper introduces a localization technique that is both simple and reliable, offering robustness and accuracy for a TLA application. The approach relies on the use of a Kalman filter and map matching. The filter incorporates a one-dimensional uniformly accelerated kinematic motion model and integrates data from various onboard sensors, including the IMU, GPS, and Electronic Control Unit (ECU), each operating at different frequencies. Moreover, the localization algorithm is integrated into the Traffic Light Advisor (TLA) system, as developed in [6]. This integration optimizes the vehicle's speed, minimizing unnecessary stops and ensuring a smoother driving experience, ultimately reducing energy consumption.

The primary contribution of this study is the development and full-scale experimental testing of a localization algorithm in urban scenarios facing GPS signal loss conditions. In particular, the algorithm utilizes multiple sensors and different sampling rates, making it suitable for implementing Advanced Driver Assistance Systems (ADAS), including Traffic Light Advisor (TLA) systems. The research mainly focuses on the urban road environment, where traditional GPS systems exhibit poor performance due to the challenges posed by urban canyoning. Instead of relying solely on velocity integration, the algorithm leverages the combined information from the Electronic Control Unit (ECU) for speed, Inertial Measurement Unit (IMU) for longitudinal acceleration, and GPS measurements to achieve a smoother output and accurate vehicle localization even in the absence of reliable GPS data. Alongside the presentation of the localization algorithm, the study also showcases the practical application of the proposed method. Specifically, experimental results for the Traffic Light Advisor (TLA) system are presented to further validate the approach introduced in [6] through real-world road tests.

3. Experimental Setup

This section aims at introducing the experimental setup used for the validation of both the Kalman filter and the Traffic Light Advisor system. The localization algorithm relies on a GPS receiver, responsible for locating the vehicle via latitude and longitude measurements, the integrated speed value available from the Electronic Control Unit (ECU) of the trolley-bus, and an Inertial Measurement Unit (IMU) that returns the values of acceleration along its axes. The main navigation system utilized is the GPS which is installed on the front part of the vehicle and provides spatial coordinates in a fixed reference frame. As mentioned, it can be missing for significant portions of the path when the number of satellites is not sufficient or the signal is not reliable, thus not allowing the algorithm to know the measurement of the vehicle's position. The information coming from the GPS receiver must be then fused with other measurements coming from the ECU, providing the longitudinal velocity of the vehicle, and the 5 DoF IMUs measuring the acceleration of a body along the three main axes (x , y , z), as well as the rotational speed around the x and y axes as shown in Figure 1.

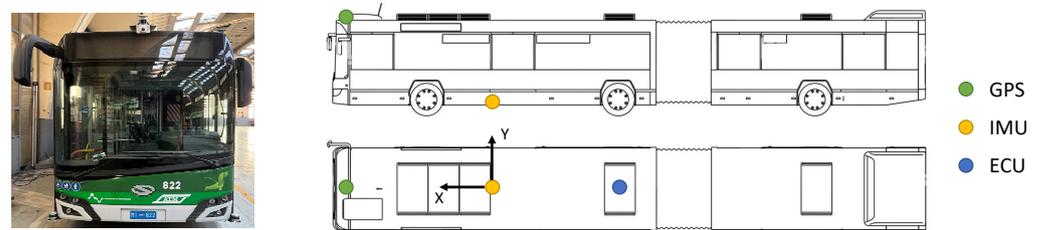


Figure 1. View and schematic representation of the instrumented vehicle used in the testing campaigns.

It is worth mentioning that, as depicted in the architectural diagram in Figure 2, the available sensors have different sampling frequencies. In fact, the GPS rate is 10 Hz, while the ECU returns the speed value at 20 Hz and the acceleration from the IMU comes at 100 Hz. As a consequence, the localization algorithm running at 100 Hz has to deal with these different frequencies.

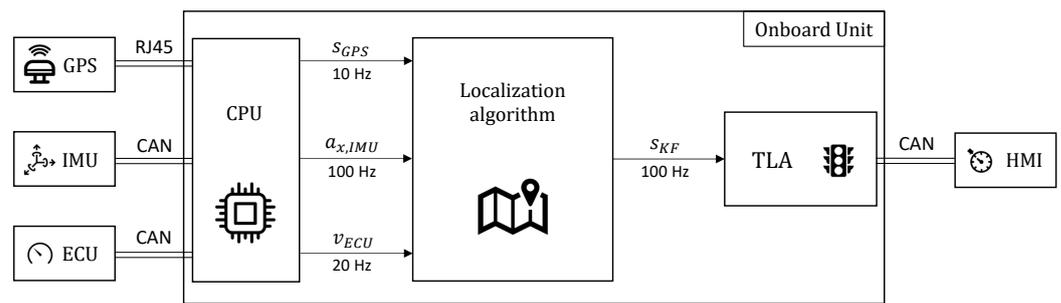


Figure 2. Architectural diagram of the vehicle’s sensor acquisition and processing.

On the onboard computational unit, the sensor acquisition, the localization algorithm, as well as the TLA system run on a soft real-time-based Robotic Operating System (ROS) architecture [32], allowing to have a simple framework for managing information coming from different sources using a publisher–subscriber logic. Within this framework, there are different nodes for publishing both the raw sensor data on the vehicle network and the vehicle data that can be read from its Controller Area Network (CAN), such as vehicle speed. This connection to the vehicle CAN-bus is used also to publish the outputs of the Traffic Light Advisor system so that the information can be shown to the driver on the integrated dashboard of the vehicle. Figure 3 reports a sample snapshot of the dedicated Human–Machine Interface (HMI). In particular, in the middle of the dashboard the driver receives a synthetic visual indication (i.e., an arrow indicator) to understand whether to accelerate or decelerate with respect to the current vehicle speed in order to reach the traffic light without stopping. This is done in order to minimize as much as possible the possible distraction source for the driver. However, additional information, such as the current traffic light status (i.e., top left corner), the time-to-change of the upcoming traffic light, and the value of the speed proposed by the TLA algorithm (i.e., bottom left corner), is provided in the periphery of the HMI.



Figure 3. Example view of the Human–Machine Interface used for TLA experimental tests (snapshot with test values for all possible outputs).

The testing area is available on a 4 km long portion of the regular service trolley-bus path in the city of Milan, being mainly covered in a preferential lane for public transportation. The route map (see Figure 4a) includes different scenarios, such as avenues with trees, urban canyoning, mid-narrow turns, and a tunnel 200 m long where the GPS is missing for a relatively long time. Furthermore, in order to have a better assessment of the localization algorithm, additional tests have been performed where a more favorable RTK correction for GPS is available, thus having a ground truth reference to evaluate the algorithm’s performances. Moreover, the second testing scenario (see Figure 4b) considered presents more severe testing conditions in terms of curve severity.

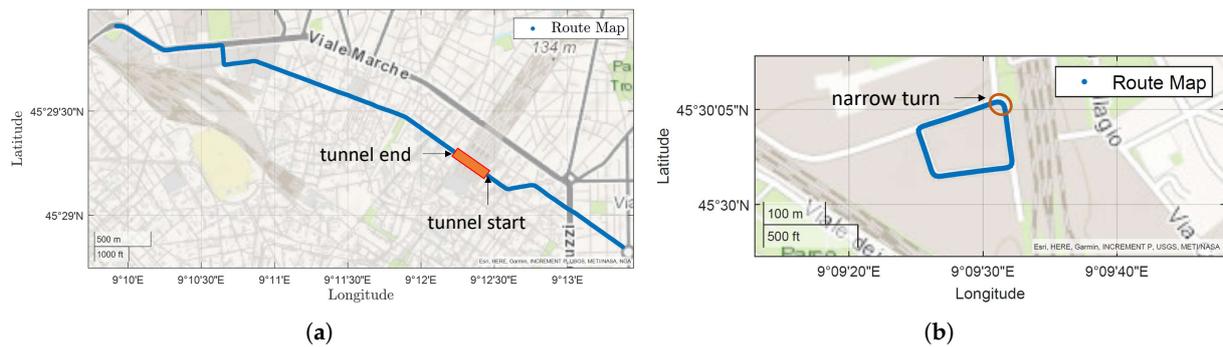


Figure 4. Testing areas in the city of Milan. (a) Testing Area 1; (b) Testing Area 2.

4. Localization Algorithm

In order to minimize the error when estimating the position, a widely spread choice while dealing with linear systems is the use of the Kalman filter, also known as the Linear Quadratic Estimator (LQE). The principle of the Kalman filter (scheme in Figure 5) is to use a dynamic model of a system, with a number of variables constituting the state vector x and describing the system itself and its evolution over time. The system prediction step is taken thanks to known input variables, i.e., control inputs u , while the available measurements z are used to properly update the propagated variables in order to minimize the difference between the predicted states and the observed quantities. For the present application, the governing equations are those related to a simple uniformly accelerated 1D model:

$$\begin{cases} s_{t+1} = s_t + v_t \cdot t + \frac{1}{2}a_t \cdot t^2 \\ v_{t+1} = v_t + a_t \cdot t \end{cases} \quad (1)$$

where s , v , and a represent the vehicle's position along the curvilinear coordinate on the path, speed, and longitudinal acceleration, respectively. The choice of such a simple model is justified by the fact that, on the one hand, the speed, and thus the accelerations, are limited. On the other hand, the TLA application needs just longitudinal accuracy along a predefined map of the path which has been obtained with the same algorithm presented in [33] based on the Cubic Hermite Spline (CHS).

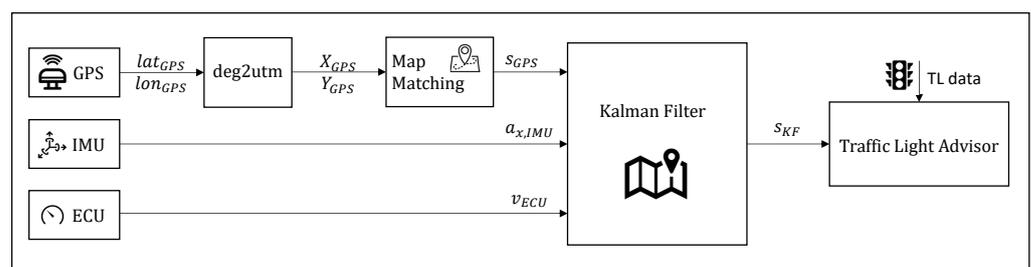


Figure 5. Scheme of the localization algorithm.

Figure 5 summarizes the scheme of the implemented Kalman filter for localizing the trolley-bus along a known map of the route followed by the vehicle. In the following, the general mathematical description of the state estimator is reported, dividing the algorithm into four stages for the sake of clarity.

4.1. Process Equation

The first step reports the process equation, defining a discrete-time linear time-varying system as

$$x_k = F_{k-1}x_{k-1} + G_{k-1}u_{k-1} + w_{k-1} \quad (2)$$

with it being possible to assume that the system matrices F_{k-1} (i.e., state transition matrix), G_{k-1} (i.e., control-input matrix), and control action u_{k-1} are known without errors. In fact, according to Equation (1), the state transition and control-input matrices can be defined as

$$F_{k-1} = \begin{bmatrix} 1 & dt \\ 0 & 1 \end{bmatrix} \quad G_{k-1} = \begin{bmatrix} \frac{1}{2}dt^2 \\ dt \end{bmatrix} \quad (3)$$

where dt stands for the integration time step equal to 0.01 s.

The process noise $w_{k-1} \sim (0, Q_{k-1})$ assumes a random Gaussian zero-mean covariance Q_{k-1} , and it accounts for the noise related both to the modeling and to the input variables. The covariance Q_{k-1} indicates how much the system model can be trusted for the prediction of the estimate. In fact, higher values of Q_{k-1} indicate lower accuracy of the model, so less weight on the estimate.

In this work, the noise due to the model is assumed to be negligible and, as an additional assumption, the covariance matrix Q_{k-1} is considered constant and diagonal:

$$Q_{k-1} = \begin{bmatrix} Q_{pos} & 0 \\ 0 & Q_{vel} \end{bmatrix} \quad (4)$$

in which the diagonal elements of the matrix are calculated as

$$Q_{pos} = dt^2 \cdot \sigma_{a_x}^2 \cdot dt^2 = 10^{-10} \quad (5)$$

$$Q_{vel} = dt \cdot \sigma_{a_x}^2 \cdot dt = 10^{-6} \quad (6)$$

where $\sigma_{a_x}^2$ represents the longitudinal acceleration variance, obtained from measurements.

4.2. Measurement Equation

The measurement equation of the system is written as

$$z_k = H_k x_k + v_k \quad (7)$$

where H_k represents the measurement matrix structured as

$$H_k = \begin{bmatrix} H_{gps} & 0 \\ 0 & H_{ecu} \end{bmatrix} \quad (8)$$

with H_{gps} and H_{ecu} being Boolean values depending on GPS and ECU data availability, as depicted in Figure 6.

As far as the H_k definition is concerned, in order to deal with the GPS outages and the multi-rate multi-sensor setup illustrated above, H_{gps} and H_{ecu} are Boolean variables defined depending on each sensor's availability at time k . Both H_{gps} and H_{ecu} are initially set to zero; if a GPS measure is available and it is considered reliable as the number of satellites received is greater than 7, then H_{gps} is set to 1. As for the speed, if a new speed measurement arrives, H_{ecu} is set to 1. It is worth noting that the speed value read from the ECU is considered always reliable, as the vehicle is thought to run in a standard adherence condition with a limited average speed. In case no measurement is available for the update, H_k remains null; thus, the state keeps on just being predicted. Performing this check every 0.01 s, the consequence, in the best-case scenario, is that the GPS update occurs just once every 10 time steps, while the ECU update happens once in 5 time steps.

Gaussian measurement noise $n_k \sim (0, R_k)$ is associated with the sensors used for the measurement update. It is characterized by zero mean and covariance R_k defined as

$$R_k = \begin{bmatrix} R_{gps} & 0 \\ 0 & R_{ecu} \end{bmatrix} \quad (9)$$

where $R_{gps} = 10^{-1}$ and $R_{ecu} = 10^{-5}$ are tuning parameters for the filter related to the reliability of the sensors, as they are obtained by computing the variance of the signals of the two sensors. In fact, more reliable sensors lead to lower \mathbf{R}_k values, while sensors introducing more noise will be responsible for higher \mathbf{R}_k values, thus leading to a lower impact of the measurement update on the state estimation. These values are then adjusted to obtain additional stability for the estimate, especially when the vehicle stands still. The idea behind the tuning is based on the following principles:

- $R_{gps} > R_{ecu}$ in the driving condition: the availability and the accuracy of the GPS depend on many different factors, while the Wheel Speed Sensor is much more reliable and accurate, as the values of the variance confirm;
- $R_{gps} \gg R_{ecu}$ in the standing-still condition: the value of R_{gps} has been increased to 1 when the vehicle speed is lower than 1 km/h; otherwise, the covariance values remain the default ones. In this way, when the vehicle stands still, the GPS data are less considered as they are less reliable and accurate for the update, while the ECU speed value becomes much more important.

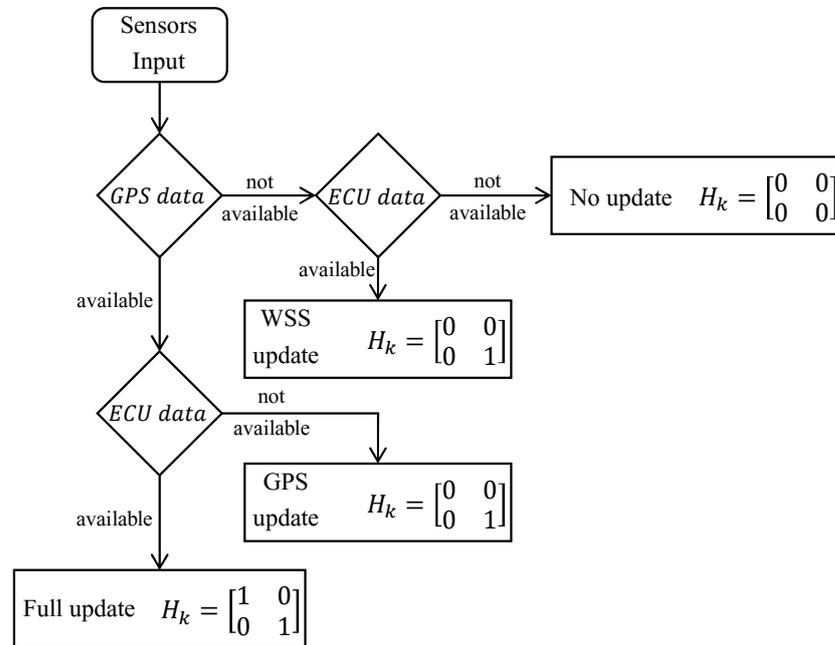


Figure 6. Measurement matrix H_k workflow definition.

4.3. Time Update Equations

In the prediction step defined in (2) at time k , the predicted state $\hat{\mathbf{x}}_k^-$ and corresponding covariance \mathbf{P}_k^- are calculated according to the model:

$$\begin{cases} \hat{\mathbf{x}}_k^- = \mathbf{F}_{k-1} \hat{\mathbf{x}}_{k-1}^+ + \mathbf{G}_{k-1} \mathbf{u}_{i-1} \\ \mathbf{P}_k^- = \mathbf{F}_{k-1} \mathbf{P}_{k-1}^+ \mathbf{F}_{k-1}^T + \mathbf{Q}_{k-1} \end{cases} \quad (10)$$

4.4. Measurement Update Equations

When the measurements are available at instant k , then the measurement step consists of the following equations:

$$\begin{cases} \mathbf{K}_k = \mathbf{P}_k^- \mathbf{H}_k^T [\mathbf{H}_k \mathbf{P}_k^- \mathbf{H}_k^T + \mathbf{R}_k]^{-1} \\ \hat{\mathbf{x}}_k^+ = \hat{\mathbf{x}}_k^- + \mathbf{K}_k [\mathbf{z}_k - \mathbf{H}_k \hat{\mathbf{x}}_k^-] \\ \mathbf{P}_k^+ = [\mathbf{I} - \mathbf{K}_k \mathbf{H}_k] \mathbf{P}_k^- \end{cases} \quad (11)$$

These equations represent the updated estimations for the state \hat{x}_k^+ and the covariance P_k^+ , respectively, and these two quantities will be required for the following step $k + 1$ in the prediction step equations shown above.

5. Traffic Light Advisor Experimental Validation

The Traffic Lights Advisor (TLA) system is an auxiliary tool for the driver, which is able to provide real-time information about the following traffic light's phase while suggesting the speed to cruise through the intersections on the path during the green light phase. Entering into the details of the TLA as an ADAS, it is expected to deal with the typical situations faced approaching a traffic light:

- Stop&Go: the algorithm is intended to properly modulate the vehicle speed in order to avoid a complete stop (when possible) in front of the traffic light.
- Last-second braking: the algorithm should inform the driver about the need to slow down, as an acceleration maneuver is not feasible.
- Unnecessary stop: the algorithm aims at suggesting to the driver the recommended speed (compliant with road limits and vehicle safety) in order to pass the upcoming intersections during the green phase of the traffic light.

As a result, the main goal of this algorithm is to save both travel time and the energy used by the vehicle. This is done by considering the 4 traffic lights ahead on the path closest to the vehicle. In particular, the algorithm is thought to consider a uniformly accelerated motion model for the vehicle to reach the suggested speed within the first traffic light of the series and then keep that velocity to safely pass the upcoming intersections without stopping. In the following, the most relevant features of the functioning of the TLA system are reported; for further details, the reader can refer to the work in [6] where the full design of the algorithm is presented.

The TLA system is based on an iterative algorithm for selecting the recommended velocity which has to fulfill the following constraints:

- The velocity has to be below the maximum allowed speed limit for the road.
- The vehicle should be able to reach the target velocity following a uniformly accelerated motion model within the end of the first available green phase of the first traffic light ahead, with the the maximum vehicle acceleration limited to 1 m/s^2 for both safety and comfort concerns.
- Considering the generic i th traffic light, the vehicle's admissible speed range is obtained from the intersection between the required velocity to reach the upcoming intersection and the admissible speed range computed for the $(i - 1)$ th traffic light.

Figure 7 represents the scheme of the Traffic Light Advisor system: the inputs, coming from onboard sensors for the localization algorithm presented in Section 4; the traffic lights; and the map enter the algorithm that produces an output that is shown in the HMI.

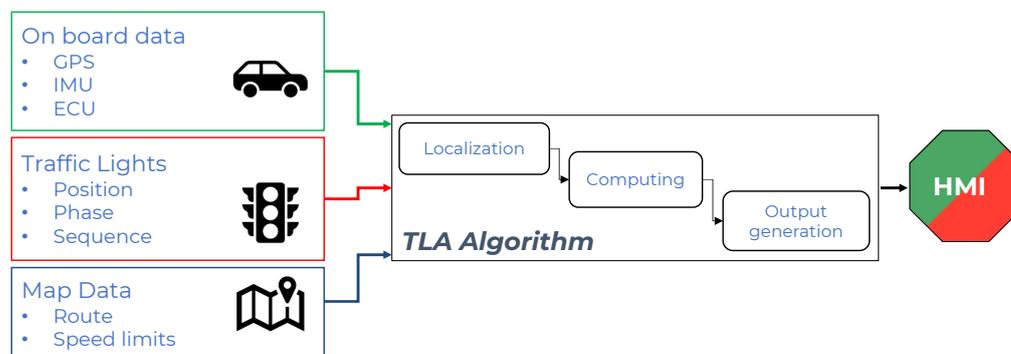


Figure 7. Schematics of the Traffic Light Advisor algorithm.

As far as the traffic light data are concerned, through the municipality of Milan there has been the possibility to know the traffic light plans of the intersections along the route.

These data are fed to a “Traffic Light Generator” which is in charge of emulating the real traffic lights on the road. This emulator, implemented in the Matlab-Simulink environment, communicates the traffic lights’ phases, sequences, and positions to the TLA algorithm.

In this work, the aim is to add an experimental validation of the non-optimal Traffic Light Advisor algorithm presented in [6]. To validate the functioning of the algorithm, similarly to what is performed in [34], three different cases are considered:

- **Base Case:** test run without the TLA used to establish a benchmark. This is obtained by letting the driver behave as usual, with no additional information available to the driver with respect to the typical driving case.
- **TLA Case:** test run with the TLA running and showing information to the driver who tries to follow the instructions. The algorithm, running in real time, conveys the information to the driver through a specifically designed HMI (see Figure 3).
- **Ideal Case:** simulation run in post-process on the basis of the TLA Case data experimentally acquired, aiming to assess the behavior of the algorithm, assuming an ideal driver able to perfectly follow the algorithm’s instructions. This is proposed to check whether the algorithm is working correctly, observing how it would operate if no constraints set by external factors such as traffic, the driver’s reflexes, and other interference were to impact its ideal behavior.

In Section 6, the results of the experimental campaign are reported, proposing a comparison between the Base Case and TLA Case. Subsequently, the real test performed on the road is compared with the aforementioned Ideal Case. The comparison is carried out not only by looking at the kinematic performances in terms of the covered distance, average speed, and acceleration but also from an energetic point of view. In fact, knowing the vehicle mass (i.e., $m = 19,800$ kg) and the vehicle acceleration a and velocity v from the onboard sensors, it is possible to obtain a rough evaluation of the power consumption expressed in (kW) starting from the inertial force as

$$P = \frac{m \cdot a \cdot v}{1000} \quad (12)$$

From the power consumption, it is possible to derive then the following index, i.e., Instantaneous Energy Consumption (IEC) expressed in (kWh/100 km), which allows quantifying the energy being drained by the vehicle:

$$IEC = P \frac{\Delta t}{\Delta s} \quad (13)$$

where Δt is the time step of the algorithm and Δs is the distance covered by the vehicle in the time span.

6. Results

This section is devoted to the presentation of the validation results of both the localization state estimator and the Traffic Light Advisor algorithm. The tests have been performed on two different paths, as shown in Figure 4, because of the availability of the traffic light information during the test runs.

6.1. Localization

The state estimator presented in Section 4 runs at 100 Hz as a standalone C++ ROS node generated in the Matlab-Simulink environment [35], subscribing to the sensors’ topics present in the ROS network of the vehicle. In the following, the results of the state estimator experimental data on the two Testing Areas are reported. It is worth noticing that the measurement used by the Kalman filter is the raw data of the GPS device, as it represents a more general and applicable condition in urban scenarios. As far as the localization accuracy is concerned, the estimation error is evaluated using as ground truth the data coming from the real-time kinematic (RTK)-corrected GPS data which reaches cm level precision.

The analysis is focused on three main conditions:

- Vehicle standing still: this condition is quite frequent in urban scenarios because of the high number of intersections and the stops along the path of a local public transportation vehicle. The typical GPS behavior in this situation is to fluctuate around the actual position, causing the vehicle localization to change, both forward and backward.
- Prolonged GPS outage: this condition is usually faced because of urban canyoning, high trees, or tunnels.
- Regular driving with curved path: this is the general scenario to be considered for the localization accuracy assessment.

The first two items are analyzed in Testing Area 1 (i.e., Figure 4a) which covers the regular service path of the vehicle. Figure 8a depicts the trend of the covered distance (i.e., the curvilinear coordinate along the path) in correspondence to a stop. As can be seen, on the one hand, the GPS trend is floating; on the other hand, the state estimator is able to provide a constant s value thanks to the increase in the R_{gps} . This allows having a constant value for the curvilinear coordinate when the vehicle is stopped, which is fundamental for the TLA application as the change in the distance of the traffic light ahead affects the TLA speed calculation. The plot reported in Figure 8b shows the state estimator trend during a prolonged GPS outage because of a 200 m long tunnel. The distance between the state estimator value s_{KF} and the GPS coordinate s_{GPS} as soon as it becomes available is equal to 1.86 m. Although this value is relatively high, the GPS benchmark after a long outage has to be considered as not reliable, as it requires some time (i.e., 20–30 m) to obtain acceptable GPS accuracy.

The testing campaign on the second Testing Area (i.e., Figure 4b) aims to evaluate the actual localization accuracy that can be obtained with the proposed localization algorithm. In fact, this area is characterized by better GPS coverage as there are no urban canyons that typically affect the GPS signal. Furthermore, the route path has four narrow turns, allowing to assess the behavior of the state estimator in the turn condition as well. In this case study, the raw data of the GPS, i.e., before the RTK correction, are fed to the localization algorithm. The state estimator output is then compared with the RTK correction GPS data which provides the ground truth value for the accuracy assessment.

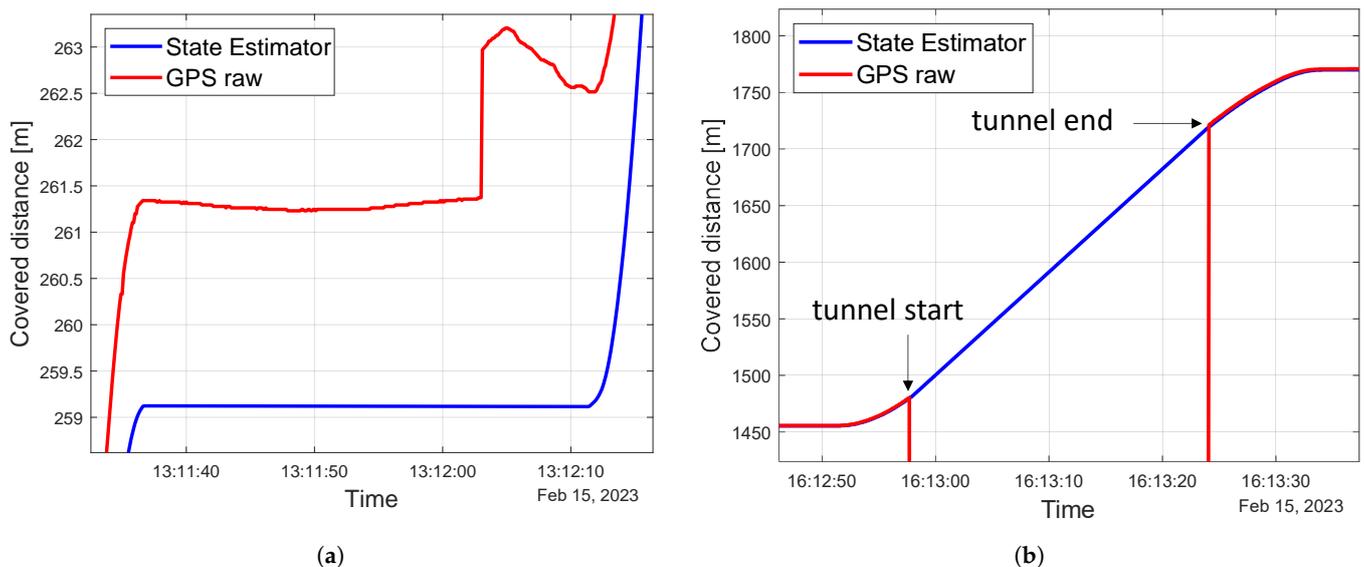


Figure 8. Testing Area 1: state estimator results analysis. (a) Vehicle standing still; (b) long GPS outage.

In Figure 9, a portion of the trend of the covered distance on the closed path of the Testing Area 2 is shown. In this scenario, it is possible to appreciate a much smoother trend of the state estimator with respect to the step-wise GPS signal because of the higher

frequency of the localization algorithm, i.e., 100 Hz. When considering the entire path, the estimation error can be calculated as

$$\epsilon_s = s_{RTK} - s_{KF} \quad (14)$$

where s_{RTK} and s_{KF} are the curvilinear coordinate along the map of the GPS RTK corrected and the proposed Kalman filter, respectively. Taking the Root Mean Square value of the error trend over the whole path, it turns out to be equal to 0.28 m. Figure 10 reports the trajectory of the vehicle estimated by the proposed Kalman filter in the narrowest turn on the path. The color indicates the estimation error ϵ_s , with the estimation error limited to 0.4 m also in the turn condition.

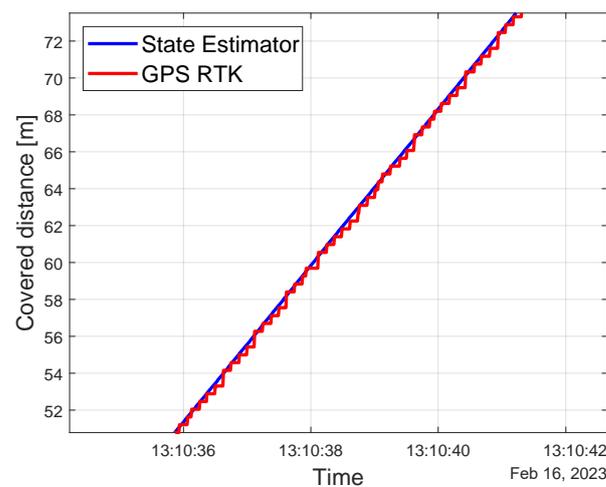


Figure 9. Testing Area 2: state estimator result analysis.

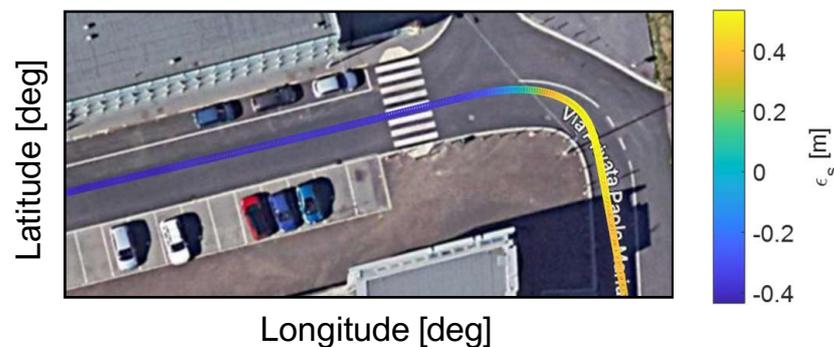


Figure 10. Testing Area 2: state estimator accuracy in curve condition.

The obtained experimental results show good accuracy for the TLA application, which just needs the vehicle location along the path to compute the distance from the upcoming traffic lights. Furthermore, as the vehicle considered is a large local public transportation vehicle, changes in speed and direction are typically not so harsh, so the choice of a one-dimensional model, on the one hand, allows to minimize the implementation effort. As far as the behavior in the curve condition is concerned, the results are promising and allow running the TLA algorithm in a general urban scenario, although the model appears to not be accurate enough for control logic dealing with vehicle lateral dynamics.

6.2. Traffic Light Advisor

As mentioned in Section 5, the assessment of the TLA performances is performed both from a kinematic and an energetic point of view. In particular, in Figure 11, the plot of the covered distance of one test run, comparing the Base Case of the driver with the TLA

Case, is shown. The horizontal lines represent the traffic lights status on the path, with the sequence between the red and green phases. It is worth mentioning that the yellow phase has been included in the red one for adding an extra safety margin to the algorithm.

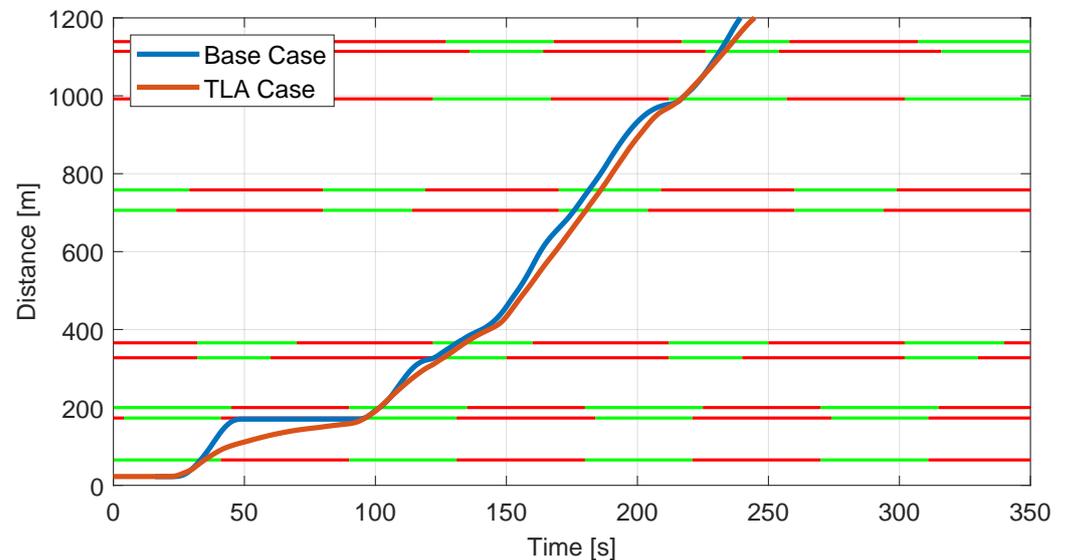


Figure 11. TLA Case comparison with respect to Base Case.

From the graph, it is possible to observe how the TLA system, besides avoiding the stop at the second traffic light encountered by the vehicle, allows having a smoother trend for the entire path, meaning that the velocity has a lower fluctuation, and thus lower acceleration.

Table 1 proposes a similar analysis from a quantitative point of view, showing the Root Mean Square values of velocity, acceleration, and IEC, previously defined. The results indicate that, when the TLA is active, the vehicle proceeds with a lower average speed with respect to the Base Case. This can sound unexpected, but it is consistent with the lower acceleration value, as the driver tends to accelerate more than required to cruise through all the intersections without stops. Besides more comfort for passengers, having lower accelerations guarantees lower energy consumption, with a 40% reduction in the energetic indicator IEC when the TLA is active. Furthermore, these values confirm the results obtained from the simulations conducted in [6].

Table 1. RMS values for speed, acceleration, and IEC: comparison between Base Case and TLA Case.

	Base Case	TLA Case	Reduction
v_{RMS} [m/s]	6.50	4.95	−23.7%
a_{RMS} [m/s ²]	0.53	0.31	−44.3%
IEC_{RMS} [$\frac{kWh}{100 km}$]	291.39	175.92	−39.6%

The plot in Figure 12 is intended to investigate in which situation the algorithm has an improvement margin and how it could be adjusted to further enhance its impact.

In fact, the spotlight is on the algorithm's reaction to the inability of the driver to perfectly follow the speed reference. The plot reports the previously shown trend of the TLA Case and it is compared with a set of ideal behaviors of the vehicle obtained by running different simulations 8 s long, each one having the vehicle position and velocity in correspondence to the start of the simulation (i.e., current actual vehicle position and velocity) as the initial conditions. This allows to see how the vehicle would have proceeded in an ideal case, thus highlighting, on the one hand, the effect of external factors as well as driver behaviors, and, on the other hand, the capability of the TLA system to adapt to

the conditions the vehicle is facing. Two main points of interest are indicated in the plot, those in which the ideal response to given real conditions (i.e., blue solid line) is suggesting a different behavior. In both cases, the vehicle should accelerate right before crossing the facing traffic light, but the driver is more confident in waiting for some seconds because the traffic in front of the vehicle is still showing a red light. This is a natural tendency of human drivers to not fully trust an indication of the HMI. In fact, although based on actual real-time data about the traffic light time-to-change, in some cases the TLA is perceived unsafe as it may also suggest accelerating when a common driver, without knowing when the light is going to change, would not.

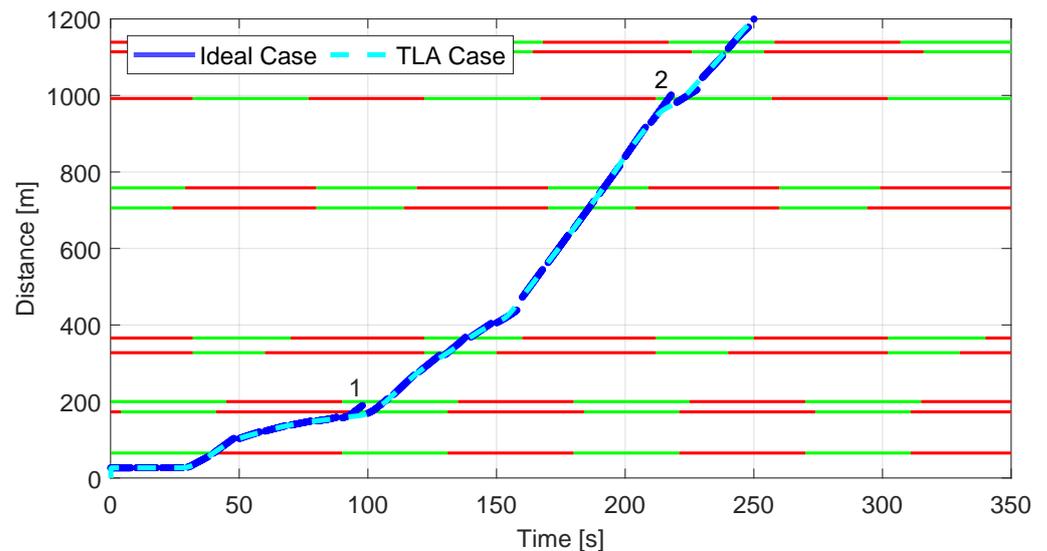


Figure 12. TLA Case comparison with respect to Ideal Case.

7. Conclusions

In this work, a localization algorithm based on Kalman filtering for a Traffic Light Advisor application and the TLA experimental validation are proposed. The implemented Kalman filter is designed to run at 100 Hz with a simple 1D kinematic model and measurements from sensors having different sampling frequencies. Real-world tests provided results accurate enough to be integrated into the TLA algorithm with an average error lower than 0.5 m, having robust behavior both in long GPS outage situations and standing-still and curve conditions thanks to the filter weight tuning and map matching. Regarding the TLA validation, the experimental campaign confirmed the positive impact on both comfort, service regularity, and energy consumption with respect to unassisted driving. The comparison with a simulated ideal case highlighted the areas of improvement for the actual implementation of the system, such as the presence of traffic in front of the vehicle, and external factors, like road unevenness, that make the driver slow down, as well as driver difficulties in following and trusting the HMI indications.

As future developments, on the one hand, it would be interesting to extend the state estimator to a 2D vehicle model, aiming at using the implemented localization algorithm for other ADAS applications also considering later dynamics of the vehicle. On the other hand, the TLA experimental validation would need additional testing campaigns to perform a larger statistical assessment of the algorithm. Furthermore, it would be valuable to introduce information about the traffic ahead in the TLA algorithm to cope with the challenges that emerged from the test.

Author Contributions: Conceptualization, E.S.; Methodology, S.A.; Software, D.V.; Validation, S.A.; Formal analysis, S.A.; Data curation, D.V.; Writing—original draft, D.V.; Writing—review & editing, D.V., S.A. and E.S.; Supervision, E.S.; Funding acquisition, E.S. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Informed consent was obtained from all subjects involved in the study.

Data Availability Statement: Not applicable.

Acknowledgments: This work is part of the project “Joint Research Lab per la Mobilità urbana”. A particular acknowledgment goes to all the partners involved in the task, ATM S.p.A., Solaris Bus & Coach sp. z o.o., AMAT, Vodafone Italia S.p.A., and Comune di Milano for supporting the project initiative.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Montanaro, U.; Dixit, S.; Fallah, S.; Dianati, M.; Stevens, A.; Oxtoby, D.; Mouzakitis, A. Towards connected autonomous driving: Review of use-cases. *Veh. Syst. Dyn.* **2019**, *57*, 779–814. [\[CrossRef\]](#)
2. Kassas, Z.Z.M.; Maaref, M.; Morales, J.J.; Khalife, J.J.; Shamei, K. Robust Vehicular Localization and Map Matching in Urban Environments through IMU, GNSS, and Cellular Signals. *IEEE Intell. Transp. Syst. Mag.* **2020**, *12*, 36–52. [\[CrossRef\]](#)
3. Stahlmann, R.; Möller, M.; Brauer, A.; German, R.; Eckhoff, D. Exploring GLOSA systems in the field: Technical evaluation and results. *Comput. Commun.* **2018**, *120*, 112–124. [\[CrossRef\]](#)
4. Barth, M.; Mandava, S.; Boriboonsomsin, K.; Xia, H. Dynamic ECO-driving for arterial corridors. In Proceedings of the 2011 IEEE Forum on Integrated and Sustainable Transportation Systems, Vienna, Austria, 29 June–1 July 2011; pp. 182–188. [\[CrossRef\]](#)
5. Cai, L.S.; Ning, G.B. Adaptive Driving Speed Guiding to Avoid Red Traffic Lights. *Appl. Mech. Mater.* **2013**, *347–350*, 3832–3836. [\[CrossRef\]](#)
6. Khayyat, M.; Gabriele, A.; Mancini, F.; Arrigoni, S.; Braghin, F. Design and Simulation-Based Testing of Connected Traffic Light Guidance Systems. *arXiv* **2022**, arXiv:2209.05783.
7. De Nunzio, G.; Canudas de Wit, C.; Moulin, P.; Di Domenico, D. Eco-driving in urban traffic networks using traffic signal information. In Proceedings of the 52nd IEEE Conference on Decision and Control, Firenze, Italy, 10–13 December 2013; pp. 892–898. [\[CrossRef\]](#)
8. Seredynski, M.; Dorransoro, B.; Khadraoui, D. Comparison of Green Light Optimal Speed Advisory approaches. In Proceedings of the 16th International IEEE Conference on Intelligent Transportation Systems (ITSC 2013), The Hague, The Netherlands, 6–9 October 2013; pp. 2187–2192. [\[CrossRef\]](#)
9. Guan, T.; Frey, C.W. Predictive fuel efficiency optimization using traffic light timings and fuel consumption model. In Proceedings of the 16th International IEEE Conference on Intelligent Transportation Systems (ITSC 2013), The Hague, The Netherlands, 6–9 October 2013; pp. 1553–1558. [\[CrossRef\]](#)
10. Suramardhana, T.A.; Jeong, H.Y. A driver-centric green light optimal speed advisory (DC-GLOSA) for improving road traffic congestion at urban intersections. In Proceedings of the 2014 IEEE Asia Pacific Conference on Wireless and Mobile, Bali, Indonesia, 28–30 August 2014; pp. 304–309. [\[CrossRef\]](#)
11. Zhang, R.; Yao, E. Eco-driving at signalised intersections for electric vehicles. *IET Intell. Transp. Syst.* **2015**, *9*, 488–497. [\[CrossRef\]](#)
12. Wegener, M.; Koch, L.; Eisenbarth, M.; Andert, J. Automated eco-driving in urban scenarios using deep reinforcement learning. *Transp. Res. Part C Emerg. Technol.* **2021**, *126*, 102967. [\[CrossRef\]](#)
13. Xu, B.; Chen, X.; Li, K.; Hu, M.; Bian, Y.; Yu, Q.; Wang, J. Double-layer speed optimization for reducing fuel consumption with vehicle-to-infrastructure communication. *J. Intell. Transp. Syst.* **2019**, *23*, 513–524. [\[CrossRef\]](#)
14. Coppola, A.; Costanzo, L.D.; Pariota, L.; Santini, S.; Bifulco, G.N. An Integrated Simulation Environment to test the effectiveness of GLOSA services under different working conditions. *Transp. Res. Part C Emerg. Technol.* **2022**, *134*, 103455. [\[CrossRef\]](#)
15. Seredynski, M.; Mazurczyk, W.; Khadraoui, D. Multi-segment Green Light Optimal Speed Advisory. In Proceedings of the 2013 IEEE International Symposium on Parallel & Distributed Processing, Workshops and Phd Forum, Cambridge, MA, USA, 20–24 May 2013; pp. 459–465. [\[CrossRef\]](#)
16. Kural, E.; Jones, S.; Parrilla, A.F.; Grauers, A. Traffic light assistant system for optimized energy consumption in an electric vehicle. In Proceedings of the 2014 International Conference on Connected Vehicles and Expo (ICCVEx), Vienna, Austria, 3–7 November 2014; pp. 604–611. [\[CrossRef\]](#)
17. Lu, Y.; Ma, H.; Smart, E.; Yu, H. Real-Time Performance-Focused Localization Techniques for Autonomous Vehicle: A Review. *IEEE Trans. Intell. Transp. Syst.* **2022**, *23*, 6082–6100. [\[CrossRef\]](#)
18. Gao, L.; Xiong, L.; Xia, X.; Lu, Y.; Yu, Z. Vehicle Localization with Vehicle Dynamics During GNSS Outages. In Proceedings of the Advances in Dynamics of Vehicles on Roads and Tracks, Gothenburg, Sweden, 12–16 August 2019; Klomp, M., Bruzelius, F., Nielsen, J., Hillemyr, A., Eds.; Springer International Publishing: Cham, Switzerland, 2020; pp. 1071–1079.
19. Gao, L.; Xiong, L.; Xia, X.; Lu, Y.; Yu, Z.; Khajepour, A. Improved Vehicle Localization Using On-Board Sensors and Vehicle Lateral Velocity. *IEEE Sens. J.* **2022**, *22*, 6818–6831. [\[CrossRef\]](#)

20. Alrouسان, Q.; Alzu'bi, H.; Pfeil, A.; Tasky, T. *Autonomous Vehicle Multi-Sensors Localization in Unstructured Environment*; SAE Technical Paper; SAE International: Warrendale PA, USA, 2020; Volume 2020. [[CrossRef](#)]
21. Gu, S.; Dai, C.; Mao, F.; Fang, W. Integration of Multi-GNSS PPP-RTK/INS/Vision with a Cascading Kalman Filter for Vehicle Navigation in Urban Areas. *Remote Sens.* **2022**, *14*, 4337. [[CrossRef](#)]
22. Bersani, M.; Mentasti, S.; Dahal, P.; Arrigoni, S.; Vignati, M.; Cheli, F.; Matteucci, M. An integrated algorithm for ego-vehicle and obstacles state estimation for autonomous driving. *Robot. Auton. Syst.* **2021**, *139*, 103662. [[CrossRef](#)]
23. Wang, F.; Zhuang, W.; Yin, G.; Liu, S.; Liu, Y.; Dong, H. Robust inter-vehicle distance measurement using cooperative vehicle localization. *Sensors* **2021**, *21*, 2048. [[CrossRef](#)]
24. Mahmoud, M.S.; Emzir, M.F. State estimation with asynchronous multi-rate multi-smart sensors. *Inf. Sci.* **2012**, *196*, 15–27. [[CrossRef](#)]
25. Kordestani, M.; Dehghani, M.; Moshiri, B.; Saif, M. A New Fusion Estimation Method for Multi-Rate Multi-Sensor Systems with Missing Measurements. *IEEE Access* **2020**, *8*, 47522–47532. [[CrossRef](#)]
26. Bai, S.; Lai, J.; Lyu, P.; Cen, Y.; Ji, B. Improved preintegration method for gnss/imu/in-vehicle sensors navigation using graph optimization. *IEEE Trans. Veh. Technol.* **2021**, *70*, 11446–11457. [[CrossRef](#)]
27. Héry, E.; Xu, P.; Bonnifait, P. Distributed asynchronous cooperative localization with inaccurate GNSS positions. In Proceedings of the 2019 IEEE Intelligent Transportation Systems Conference (ITSC), Auckland, New Zealand, 27–30 October 2019; pp. 1857–1863. [[CrossRef](#)]
28. Wang, F.; Xu, L.; Zhuang, W.; Yin, G.; Pi, D.; Liang, J.; Liu, Y.; Lu, Y. Geometry-Based Cooperative Localization for Connected Vehicle Subject to Temporary Loss of GNSS Signals. *IEEE Sens. J.* **2021**, *21*, 23527–23536. [[CrossRef](#)]
29. Korb, M.; Huang, Q.; Stockel, P.; Kappen, G.C.; Weber, B.; Garcia, M. A Cellular-Modem-Hosted Low-Cost Single-Shot Dual-Mode Assisted-GNSS Receiver for the Internet of Things. In Proceedings of the 2020 IEEE/ION Position, Location and Navigation Symposium (PLANS), Portland, OR, USA, 20–23 April 2020; pp. 1273–1279. [[CrossRef](#)]
30. Zandbergen, P.A. Accuracy of iPhone Locations: A Comparison of Assisted GPS, WiFi and Cellular Positioning. *Trans. GIS* **2009**, *13*, 5–25. [[CrossRef](#)]
31. Sheng, L.; Tian, R.; Ye, H. Assist GPS to Improve Accuracy Under Complex Road Conditions Using Sensors on Smart Phone. In Proceedings of the MLICOM 2020: International Conference on Machine Learning and Intelligent Communications, Shenzhen, China, 26–27 September 2020; Springer International Publishing: Cham, Switzerland, 2021; pp. 287–303.
32. Quigley, M.; Gerkey, B.; Conley, K.; Faust, J.; Foote, T.; Leibs, J.; Berger, E.; Wheeler, R.; Ng, A. ROS: An open-source Robot Operating System. In Proceedings of the IEEE International Conference on Robotics and Automation (ICRA) Workshop on Open Source Robotics, Kobe, Japan, 12–17 May 2009.
33. Arrigoni, S.; Braghin, F.; Cheli, F. MPC trajectory planner for autonomous driving solved by genetic algorithm technique. *Veh. Syst. Dyn.* **2022**, *60*, 4118–4143. [[CrossRef](#)]
34. Chen, H.; Rakha, H.A. Developing and Field Testing a Green Light Optimal Speed Advisory System for Buses. *Energies* **2022**, *15*, 1491. [[CrossRef](#)]
35. The MathWorks, Inc. Generate a Standalone ROS Node from Simulink. Available online: <https://it.mathworks.com/help/ros/ug/generate-a-standalone-ros-node-from-simulink.html> (accessed on 31 July 2023).

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.