Angle of Arrival-Based Cooperative Positioning for Smart Vehicles

Alessio Fascista, Giovanni Ciccarese, Angelo Coluccia, *Senior Member, IEEE*, and Giuseppe Ricci, *Senior Member, IEEE*

***Abstract*— The limited localization capabilities provided by global navigation satellite systems (GNSS) is one of the main obstacles toward the development of reliable road safety applications in urban scenarios. In order to improve GNSS accuracy, a number of approaches have been proposed which exploit additional position-related information, for instance provided by local inertial sensors. However, such solutions cannot meet the very stringent accuracy requirements of safety applications, which call for advanced processing and the fusion of position-related signals and data from heterogeneous sources. In this paper, we aim at combining the potential of antenna array processing with a suitably-designed cooperation strategy that exploits vehicle-to-vehicle and vehicle-to-infrastructure communications. Particularly, we define a novel tracking algorithm with asynchronous updates triggered by beacon packet receptions, from which angle of arrival estimates are opportunistically obtained. A dynamic setting of relevant parameters allows the resulting cooperative positioning algorithm to adapt to the different operating conditions found in urban vehicular contexts. Simulation results under realistic environment conditions show that the proposed algorithm can achieve high position accuracy even in sparse scenarios, outperforming a natural competitor while keeping lightweight communication and low computational complexity.**

***Index Terms*— Extended Kalman filter (EKF), vehicular ad-hoc network (VANET), vehicle-to-infrastructure (V2I) and vehicle-to-vehicle (V2V) communications, angle of arrival (AOA) estimation.**

I. Introduction

ADVANCED vehicular technologies are an emerging trend of great interest for both academia and industry. In shifting from traditional to smart or even autonomous vehicles, signal processing is playing an increasingly important role [1], [2]. In particular, road safety applications — which are at the top of the future intelligent transportation systems (ITS) development agenda — pose numerous challenges for which ultimate solutions are still unavailable. One of the most important issues is how to guarantee accurate position information, which typically translates into sub-meter precision requirements, in all the very diverse situations that are found in vehicular scenarios [3]. Global navigation satellite systems (GNSS), in particular the global positioning system (GPS), are currently the leading technology for vehicular localization and navigation; however, recent studies show that the delivered accuracy cannot always meet the very stringent requirements of crucial position-based applications [4], particularly in dense urban environments, because of satellite visibility interruption, vehicle dynamics, and local causes (e.g., receiver noise, multipath) [5].

Early work on the topic focused on the possibility to improve the GPS localization performance by exploiting additional information available from on-board kinematic sensors (e.g., odometers, accelerometers, gyroscopes, etc.), so integrating satellite positioning with inertial navigation (INS) [6]–[10]. This kind of approaches are referred to as *standalone* since the vehicle uses only locally available information in the localization process, without any form of interaction.

Opportunistic use of other wireless communication systems that are deployed in the same area has been proposed as a ready cost-effective solution to increase the availability of the positioning service [11]–[18]. For instance, in [18] a pedestrian dead reckoning system, that combines inertial devices and sensor network technology, has been proposed. More generally, wireless localization solutions are currently receiving great attention also by the mobile "apps" industry because they provide location awareness at zero-cost when GPS is unreliable or even switched-off [19]–[21]. However, the sub-meter accuracy requirements of road safety applications cannot be met by means of the coarse information available from such systems [22], [23]. It is foreseen that advanced signal processing will play a major role in designing future localization systems for the vehicular context; such systems, in fact, will need data fusion and dynamic filtering techniques to process all the available position-related information from heterogeneous sources in order to accurately operate also in GPS-denied environments [24].

In recent years vehicular ad-hoc networks (VANETs) [25], [26] have been proposed by the automotive research community as a mean to realize a connected road environment where vehicles and infrastructure components can communicate in a tailored way via dedicated protocols, with differentiated quality-of-service (QoS) according to the application (i.e., safety or comfort), to implement a variety of services including more accurate positioning [27]. Specifically, to meet the requirements of road safety applications, cooperative positioning (CP) approaches have been defined [28]: basically,

Manuscript received March 27, 2017; revised July 1, 2017 and September 19, 2017; accepted October 24, 2017. Date of publication November 22, 2017; date of current version September 7, 2018. The Associate Editor for this paper was S. Kong. *(Corresponding author: Angelo Coluccia.)*

The authors are with the Department of Innovation Engineering, University of Salento, 73100 Lecce, Italy (e-mail: alessio.fascista@unisalento.it; giovanni.ciccarese@unisalento.it; angelo.coluccia@unisalento.it; giuseppe.ricci@unisalento.it).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TITS.2017.2769488

a CP system aims at improving the accuracy by jointly processing position-related data exchanged among a group of VANET nodes.

Different cooperation schemes have been investigated by the research community. Some CP approaches try to augment the GPS positioning performance by exploiting additional signals, for instance range estimates based on received signal strength (RSS), time of arrival (TOA), and time difference of arrival (TDOA) [29], [30], possibly coupled with node velocities. Particularly, [29] addresses cooperative positioning of a cluster of vehicles resorting to an extended Kalman filter (EKF) fed by ranges between nodes, computed from RSS measurements, and node velocities; in addition, the algorithm forces the position estimates to be within the road boundaries. Similarly, in [30] the EKF is fed by GPS measurements and ranges between nodes computed from TDOA measurements of received packets. Map information is also used to improve localization accuracy. A different technique that exploits exchanged GPS raw data in order to mitigate the influence of degraded satellite signals is presented in [31]. It relies on pseudoranges GPS data for participating vehicles that are fed to a Kalman filter (KF), however does not consider any inter-vehicles measurements. Enhanced performance can be obtained by adding INS and GPS Doppler shift data [32]. Along the same idea, two Bayesian approaches that make use of pseudorange differences are proposed in [33].

It has been noticed that the achievable accuracy for most of the methods discussed above is limited by the insufficient information content of the considered signal: in particular, in [28] it is analytically proved that neither RSS nor TOA/TDOA ranging methods can provide a positioning accuracy below one meter as required for safety applications. More specifically, RSS is too sensitive to uncertainties on the path-loss exponent, exhibiting high ranging errors even for small estimation errors. As regards TOA/TDOA techniques, they suffer from the lack of an accurate synchronization. We will come back on this point in Sec. II-E. This issue is particularly critical in *GPS-free* contexts [34]–[37], where only terrestrial signals can be exploited for localization in challenging conditions (e.g., urban canyons).

To solve the impasse, the research community is recently steering towards the adoption of more informative signals and, at the same time, more advanced cooperation strategies. In particular, in [34] a cooperative inertial navigation (CIN) method is proposed in a vehicle-to-vehicle (V2V) scenario, where standalone INS-based positioning is improved by fusing local data with (INS-based) position estimates provided by vehicles traveling in the opposite direction. To this aim, the zero Doppler shift condition is exploited to detect vehicles passage times. The same idea has been applied in [35] in a vehicle-to-infrastructure (V2I) scenario where communication packets are exploited to estimate vehicle positions with lane-level accuracy: two road-side units (RSUs) placed on the opposite sides of the road broadcast their position and road geometry information; vehicles use such a broadcast information, together with odometer data and Doppler shift of the received signals, to compute their lane positions on the road. Still, the achievable accuracy is insufficient for safety-critical vehicular applications.

A leap forward in this respect is the use of smart antennas, which has been suggested to improve VANET communication performance via beamforming [38]. This additional equipment on-board, in particular the presence of a uniform linear array (ULA), can be in fact exploited also for localization purposes [39], [40]. Fascista *et al.* [39] have investigated a V2I scenario where angle of arrival (AOA) estimates — obtained by the multiple signal classification (MUSIC) algorithm [41] on the basis of beacon packets broadcast by an RSU in known position — are fed, together with local INS measurements, to a weighted least squares (WLS) localization algorithm. This approach turns out to outperform GPS in urban environments. In [40], self-localization of vehicles is performed implementing an EKF that is fed by AOA and round-trip time-of-flight (RTOF) measurements, in addition to (locally measured, i.e., by the vehicle) velocity and yaw rate. To this end, cooperative RSU (therein denoted RF-based landmarks) are introduced: such RF-based landmarks respond to vehicles' requests making it possible to measure AOA and RTOF.

Motivated by such results, we propose a recursive algorithm (notice that the one in [39] is not recursive) for a more general scenario than the one considered in [40]. In fact, in addition to V2I communications, also vehicle-to-vehicle (V2V) may be present, involving a variable number of vehicles and different operational conditions. In principle, both AOA and RTOF measurements, as in [40], might be considered; however, it is expected that RTOF measurements in a cooperative scenario would pose serious problems for the medium access control, see also [28]. We aim at bridging the potential of a low-level (PHY-layer) signal processing with a carefully-designed cooperation strategy at higher-level (VANET) to obtain a full positioning system able to automatically adapt to the different situations in a dynamic way, and relying on a minimal data exchange. For this more general scenario, we design a tracking algorithm based on a proper modification of the classical EKF approach, with asynchronous updates triggered by beacon packet receptions and adaptive control of the covariance matrices. The analysis in a realistic urban scenario demonstrates that the proposed algorithm can satisfy the very stringent requirements of road safety applications, even with a small number of cooperating nodes, and is able to cope with the different operating conditions at play in real-world vehicular contexts. Our approach is very effective also in sparse scenarios while keeping a lightweight communication and a low computational complexity, whereas most CP approaches need a dense network topology and heavily rely on infrastructure and/or inter-vehicle cooperation, with significant communication overhead in the VANET.

The remaining of the paper is organized as follows. In Sec. II we describe in details the design of the algorithm, including its operating modes and settings. Then, in Sec. III we analyze our solution, assessing the performance by means of simulations in different realistic urban scenarios, also in comparison with the most similar (to the best of our knowledge) algorithm that can be found in the literature. Based on the outcomes of this analysis, in Sec. IV we provide a

possible greedy strategy to select the cooperating vehicles. Finally, some conclusions are given in Sec. V.

II. Algorithm Design

A. Reference Scenario and System Architecture

We consider, as reference scenario, an urban area where some of the vehicles are equipped with an IEEE 802.11p radio device, thus enabling VANET communications. Moreover, we assume that RSUs with the same communication capabilities may be located in fixed, known positions along the roadsides and at the intersections. As will be shown in Sec. III, it is not required a dense deployment nor an intense vehicular traffic.

All such VANET nodes (either vehicles or RSUs) periodically send beacon packets in broadcast. A vehicle that is running the CP algorithm, referred to as *U-vehicle*, is equipped also with a ULA to estimate the AOA of impinging packets. Therefore, the role of each beacon packet received by a U-vehicle is twofold: on the one hand, it is used to obtain an AOA estimate based on the received signal; on the other hand, it carries, among other information, position and velocity estimates of the transmitting node, if it is a vehicle, or its exact position if it is an RSU. This information, together with the local velocity reading, obtained from the onboard INS, see for instance [42], may be combined by the tracking algorithm in order to improve the U-vehicle position estimation. It is worth noting that the cooperating vehicles can be, in turn, U-vehicles executing a local instance of the algorithm yielding their broadcast estimates, but they can also be ordinary VANET nodes that broadcast their GPS position and velocity readings — or any better estimates obtained by means of methods like those described in Sec. I. In the following, to expose the algorithm, we focus on a single U-vehicle receiving information from other vehicles and RSUs, irrespective of the way the cooperating vehicles are estimating their positions.

The filtering process is, in general, asynchronous and is based on the iteration of two phases: a *prediction* phase and an *update* phase. In particular, as depicted in Fig. 1, the filter can be triggered either by a fixed local clock or by the reception of a new beacon packet. The clock ticks at the INS reading rate f_{INS} and the filter is updated using only the local velocity information of the U-vehicle. Similarly to what happens in common GPS/INS integration methods, we exploit such standalone INS measurements in order to improve the localization system reliability and to avoid a fast accuracy degradation caused by the lack of received beacon packets for some time interval. Reception of a beacon packet, instead, allows the filter to update its state using a new AOA measurement, an anticipated INS reading (of the U-vehicle velocity) and, in case the beacon has been transmitted by a vehicle, the estimate of its velocity. In addition, any event occurring during the time in which the system is busy for processing will be ignored.[1] This asynchronous update makes sense also in presence of delay jitter and packet losses.

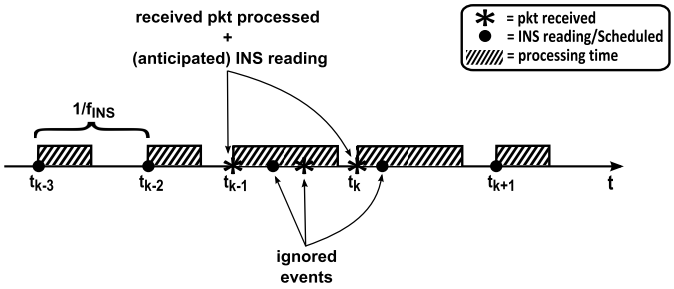


Fig. 1. A pictorial description of the selection process of update time instants.

B. Prediction Phase

The prediction phase, which is the first processing stage, performs a time alignment between the information contained in the state — whose last update occurred at t_{k-1} — and the information available at the current instant t_k. Formally, the state vector at t_{k-1} contains information related to a certain number n_{k-1} of tracked vehicles; more precisely, it contains the 2D cartesian position and velocity components of each vehicle, $V_i, i = 1, \ldots, n_{k-1}$. Thus, letting

$$\boldsymbol{x}_{k-1}^{(i)} = \left[p_x(V_i, t_{k-1})\ v_x(V_i, t_{k-1})\ p_y(V_i, t_{k-1})\ v_y(V_i, t_{k-1})\right] \tag{1}$$

where $p_\eta(V_i, t_{k-1})$ and $v_\eta(V_i, t_{k-1})$ are the position and velocity components along the η axis of the vehicle V_i, respectively, it follows that the state vector at t_{k-1} is defined as

$$\boldsymbol{x}_{k-1} = \left[\boldsymbol{x}_{k-1}^{(1)}\ \boldsymbol{x}_{k-1}^{(2)} \ldots\ \boldsymbol{x}_{k-1}^{(n_{k-1})}\right]^T. \tag{2}$$

The $V_i, i = 1, \ldots, n_{k-1}$, form a set $\mathcal{V}_{k-1} \subset \mathcal{I}$, where $\mathcal{I}$ is the set of all the possible unique network identifiers (IDs) and V_1 is the ID of the U-vehicle (i.e., $\boldsymbol{x}_{k-1}^{(1)}$ represents the part of the state associated to the U-vehicle). Each V_i has an associated aging time, say T_{V_i}, $2 \le i \le n_{k-1}$,[2] that accounts for the time elapsed from the last received beacon packet. As to ^{T}, it denotes the transpose operator. The way the n_{k-1} vehicles are selected and dynamically updated over time will be discussed in Sec. IV. It is also worth noting that RSUs are not part of the filter state since their position is fixed and known. However, each RSU R has an associated aging time, say T_R: active RSUs are the ones whose aging time is less than a maximum preassigned value.

Conversely, for the kinematics of vehicles the proposed algorithm assumes, according to a common land vehicle navigation framework [43], a nearly constant velocity model

$$\boldsymbol{x}_k = \boldsymbol{F}\boldsymbol{x}_{k-1} + \boldsymbol{w}_{k-1} \tag{3}$$

where

$$\boldsymbol{F} = \text{diag}(\boldsymbol{F}_b, \ldots, \boldsymbol{F}_b) \tag{4}$$

[1] Notice that in operational implementations on typical onboard processors, the processing time for a Markovian (one-lag) discrete-time filter is small, hence the probability of receiving a new packet during such an interval is negligible. When this is not the case, it means that there is a significant load on the network, hence anyway discarding some packets does not produce a considerable loss since new ones will be likely received soon.

[2] As previously mentioned, the algorithm always tracks the U-vehicle on which it is running.

is the $4n_{k-1} \times 4n_{k-1}$ state transition matrix with

$$\boldsymbol{F}_b = \begin{bmatrix} 1 & (t_k - t_{k-1}) & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & (t_k - t_{k-1}) \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{5}$$

and $\boldsymbol{w}_{k-1}$ is the process noise accounting for variations on the mobility model. The $\boldsymbol{w}_{k-1}$s are assumed to be independent random vectors over time; in particular, $\boldsymbol{w}_{k-1}$ is a zero-mean Gaussian random vector having covariance matrix

$$\boldsymbol{Q} = \mathrm{diag}(\boldsymbol{Q}_b, \ldots, \boldsymbol{Q}_b) \tag{6}$$

where

$$\boldsymbol{Q}_b = q \begin{bmatrix} \frac{(t_k - t_{k-1})^3}{3} & \frac{(t_k - t_{k-1})^2}{2} & 0 & 0 \\ \frac{(t_k - t_{k-1})^2}{2} & (t_k - t_{k-1}) & 0 & 0 \\ 0 & 0 & \frac{(t_k - t_{k-1})^3}{3} & \frac{(t_k - t_{k-1})^2}{2} \\ 0 & 0 & \frac{(t_k - t_{k-1})^2}{2} & (t_k - t_{k-1}) \end{bmatrix} \tag{7}$$

is obtained by discretizing a continuous white noise acceleration model with spectral density of level q $[m^2/s^3]$ [44], [45]. Notice that q is a design parameter that must be set according to the changes in the velocity over the sampling interval. Its setting is given in Sec. III.

The prediction phase actualizes at t_k the state estimate $\hat{\boldsymbol{x}}_{k-1|k-1}$ available from the previous update (at t_{k-1}) according to the formula

$$\hat{\boldsymbol{x}}_{k|k-1} = \boldsymbol{F}\hat{\boldsymbol{x}}_{k-1|k-1} \tag{8}$$

$$\boldsymbol{P}_{k|k-1} = \boldsymbol{F}\boldsymbol{P}_{k-1|k-1}\boldsymbol{F}^T + \boldsymbol{Q} \tag{9}$$

being $\boldsymbol{P}_{k|k-1}$ and $\boldsymbol{P}_{k-1|k-1}$ the error covariance matrices associated with $\hat{\boldsymbol{x}}_{k|k-1}$ and $\hat{\boldsymbol{x}}_{k-1|k-1}$, respectively.

C. Update Phase

The update phase is performed by exploiting the new measurements available at t_k. As previously mentioned, the type of information depends on the event that triggered the filtering step; in particular, it may consist of the local velocity read from the onboard INS, the AOA of the received beacon packet, and the cooperating vehicle's velocity estimate included within.[3] Notice that, in principle, any GPS information about the U-vehicle position possibly available at t_k can also be filtered in the same update step. However, in the considered urban scenario this information is inaccurate or even completely unreliable, which in fact motivates our work on cooperative positioning aimed at improving the localization accuracy of GPS. On the other hand, one may also consider exchanging additional information in the beacon packets to implement more involved cooperation schemes. Since our aim is to demostrate that a high level of accuracy can be achieved by selecting a few information from the VANET, in this work we stick to the basic set of information listed above, which are the minimum necessary for implementing the proposed localization algorithm, leaving possible extensions to future works.

The update of the filter state at the current time instant t_k is performed on the basis of one of the four different cases listed below.

1) Case 1: a beacon packet has been received from a vehicle which is part of the selected vehicles in the set $\mathcal{V}_{k-1}$;
2) Case 2: a beacon packet has been received from an RSU;
3) Case 3: the local INS reading clock has ticked;
4) Case 4: a beacon packet has been received from a vehicle which is *not* part of the selected vehicles in the set $\mathcal{V}_{k-1}$.

Case 1. A beacon packet has been received from a vehicle which is part of the selected vehicles in the set $\mathcal{V}_{k-1}$.

In particular, recall that $2 \leq i \leq n_{k-1}$ is the index of the corresponding components in the block partition of the state vector as from eq. (2). In this case, the set of selected vehicles $\mathcal{V}_k$ to be exploited at the current time instant t_k remains the same of the previous step, i.e., $\mathcal{V}_k = \mathcal{V}_{k-1}$. The velocity components of both vehicles — one obtained by local INS reading, the other one extracted from the payload of the received packet — and the AOA estimate are concatenated to build the whole measurement vector, expressed as

$$\boldsymbol{z_k} = \boldsymbol{h}_k(\boldsymbol{x}_k) + \boldsymbol{v}_k \tag{10}$$

where the $\boldsymbol{h}_k(\boldsymbol{x}_k)$ mapping includes both linear and nonlinear functions. The first ones are associated with velocity components whereas the second one is associated with the AOA estimation. Therefore, we can split $\boldsymbol{h}_k(\boldsymbol{x}_k)$ as

$$\boldsymbol{h}_k(\boldsymbol{x}_k) = \begin{bmatrix} \boldsymbol{h}_k^{\mathrm{L}}(\boldsymbol{x}_k) \\ \boldsymbol{h}_k^{\mathrm{NL}}(\boldsymbol{x}_k) \end{bmatrix} \tag{11}$$

where

$$\boldsymbol{h}_k^{\mathrm{L}}(\boldsymbol{x}_k) = \left[v_x(V_1, t_k)\ v_y(V_1, t_k)\ \tilde{v}_x(V_i, t_k)\ \tilde{v}_y(V_i, t_k) \right]^T \tag{12}$$

and

$$\boldsymbol{h}_k^{\mathrm{NL}}(\boldsymbol{x}_k) = \arccos\left(\boldsymbol{r}_u^T(V_i, t_k)\boldsymbol{v}_u(V_1, t_k) \right) \tag{13}$$

where $\boldsymbol{r}_u(V_i, t_k)$ and $\boldsymbol{v}_u(V_1, t_k)$ are unit vectors obtained from the normalization of the difference vector $\boldsymbol{r}(V_i, t_k) = \left[\boldsymbol{p}(V_i, t_k) - \boldsymbol{p}(V_1, t_k) \right]$ and the U-vehicle velocity vector $\boldsymbol{v}(V_1, t_k) = \left[v_x(V_1, t_k)\ v_y(V_1, t_k) \right]^T$, respectively (see Fig. 2).

It is well known that the Kalman filter is optimal for Gaussian noises and a linear mapping between state variables $\boldsymbol{x}_k$ and the measurements vector $\boldsymbol{z_k}$. Here, as previously highlighted, $\boldsymbol{h}_k^{\mathrm{NL}}(\boldsymbol{x}_k)$ is not linearly related to the state. Therefore, we must resort to an EKF. As regarding $\boldsymbol{v}_k$, it is a zero-mean Gaussian random vector representing the measurements noise

[3] It should be noticed that there exists a discrepancy between the current time instant t_k, which corresponds to the timestamp of the packet at the receiver side, and the time instant at which the received kinematic data are associated, timestamped at the transmitter side. However, this difference is generally so small that the velocity information can be considered the same. Therefore, in the following we will neglect it.

with covariance matrix

$$\boldsymbol{R}_k = \begin{bmatrix} \sigma^2_{v_x}(V_1,t_k) & 0 & 0 & 0 & 0 \\ 0 & \sigma^2_{v_y}(V_1,t_k) & 0 & 0 & 0 \\ 0 & 0 & \tilde{\sigma}^2_{v_x}(V_i,t_k) & 0 & 0 \\ 0 & 0 & 0 & \tilde{\sigma}^2_{v_y}(V_i,t_k) & 0 \\ 0 & 0 & 0 & 0 & \sigma^2_{\theta_k} \end{bmatrix} \tag{14}$$

being $\sigma_{v_x}(V_1,t_k)$, $\sigma_{v_y}(V_1,t_k)$, $\tilde{\sigma}_{v_x}(V_i,t_k)$, and $\tilde{\sigma}_{v_y}(V_i,t_k)$ the standard deviations of the velocity measurements at t_k along the x and y directions for both the U-vehicle and the vehicle V_i, respectively. In particular, $\tilde{\sigma}_{v_x}(V_i,t_k)$ and $\tilde{\sigma}_{v_y}(V_i,t_k)$ are extracted from the received packet. Hereafter, we denote with $\tilde{z}$ the estimate of z carried by the packet. As to σ_{θ_k}, it denotes the standard deviation related to the AOA estimation. A proper dynamic setting of such a value will be described in Sec. II-D.

Finally, the update phase is performed as

$$\hat{\boldsymbol{x}}_{k|k} = \hat{\boldsymbol{x}}_{k|k-1} + \boldsymbol{K}_k\left(z_k - \boldsymbol{h}_k(\hat{\boldsymbol{x}}_{\boldsymbol{k|k-1}})\right) \tag{15}$$

$$\boldsymbol{P}_{k|k} = \left(\boldsymbol{I}_{4n_{k-1}\times 4n_{k-1}} - \boldsymbol{K}_k\hat{\boldsymbol{H}}_k\right)\boldsymbol{P}_{k|k-1} \tag{16}$$

where $\boldsymbol{I}_{N\times N}$ denotes the $N\times N$ identity matrix and

$$\hat{\boldsymbol{H}}_k = \left.\frac{\partial \boldsymbol{h}_k(\boldsymbol{x})}{\partial \boldsymbol{x}}\right|_{\boldsymbol{x}=\hat{\boldsymbol{x}}_{k|k-1}} \tag{17}$$

is the Jacobian matrix of $\boldsymbol{h}_k(\boldsymbol{x})$ evaluated at the current state estimation. Using eq. (17), we can rewrite $\hat{\boldsymbol{H}}_k$ as

$$\hat{\boldsymbol{H}}_k = \begin{bmatrix} \hat{\boldsymbol{H}}^{\text{L}}_k \\ \hat{\boldsymbol{H}}^{\text{NL}}_k \end{bmatrix} \tag{18}$$

where, denoting by $\otimes$ the Kronecker product,

$$\hat{\boldsymbol{H}}^{\text{L}}_k = \boldsymbol{A}\otimes\boldsymbol{B} \tag{19}$$

with the block $\boldsymbol{B}$ defined as

$$\boldsymbol{B} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{20}$$

and the block $\boldsymbol{A}$ given by

$$\boldsymbol{A} = \begin{bmatrix} 1 & \boldsymbol{0}_{1\times(i-2)} & 0 & \boldsymbol{0}_{1\times(n_{k-1}-i)} \\ 0 & \boldsymbol{0}_{1\times(i-2)} & 1 & \boldsymbol{0}_{1\times(n_{k-1}-i)} \end{bmatrix} \tag{21}$$

and $\boldsymbol{A} = \boldsymbol{I}_{2\times 2}$ for $i > 2$ and $i = 2$, respectively. As to $\boldsymbol{0}_{n\times m}$, it denotes an $n\times m$ matrix of zeros.

$\hat{\boldsymbol{H}}^{\text{NL}}_k$ can be expressed as

$$\hat{\boldsymbol{H}}^{\text{NL}}_k = \left.\frac{\partial \boldsymbol{h}^{\text{NL}}_k(\boldsymbol{x})}{\partial \boldsymbol{x}}\right|_{\boldsymbol{x}=\hat{\boldsymbol{x}}_{k|k-1}}. \tag{22}$$

Explicit computation of its entries is tedious, but straightforward.

Finally, the Kalman gain is computed as

$$\boldsymbol{K}_k = \boldsymbol{P}_{k|k-1}\hat{\boldsymbol{H}}^T_k\left(\hat{\boldsymbol{H}}_k\boldsymbol{P}_{k|k-1}\hat{\boldsymbol{H}}^T_k + \boldsymbol{R}_k\right)^{-1}. \tag{23}$$

This derivation can be interpreted as an asynchronous *semi-extended* Kalman filter. In practice, the filter behaves like a KF for linear data (i.e., velocities) whereas it acts like an EKF for nonlinear data (i.e., AOAs).

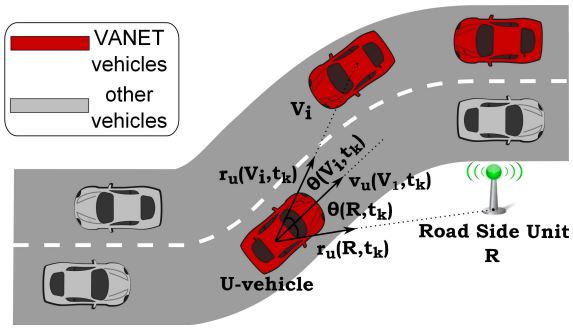


Fig. 2. A graphical representation of the angle of arrival geometry.

Case 2. A beacon packet has been received from an RSU.

As previously mentioned, RSU nodes are not part of the EKF state since their positions are fixed and known. Thus, also in this second case, $\mathcal{V}_k = \mathcal{V}_{k-1}$. The algorithm steps are the same as eqs. (8), (9), (15), and (16), where the linear mapping $\boldsymbol{h}^{\text{L}}_k(\boldsymbol{x}_k)$ simply reduces to the own INS velocity reading, i.e.,

$$\boldsymbol{h}^{\text{L}}_k(\boldsymbol{x}_k) = \left[v_x(V_1,t_k)\ v_y(V_1,t_k)\right]^T \tag{24}$$

while the nonlinear mapping $\boldsymbol{h}^{\text{NL}}_k(\boldsymbol{x}_k)$, which models the AOA, includes the fixed and known RSU's position $\boldsymbol{p}(R) = \left[p_x(R)\ p_y(R)\right]^T$ and is expressed as

$$\boldsymbol{h}^{\text{NL}}_k(\boldsymbol{x}_k) = \arccos\left(\boldsymbol{r}^T_u(R,t_k)\boldsymbol{v}_u(V_1,t_k)\right) \tag{25}$$

where now $\boldsymbol{r}_u(R,t_k)$ is the unit vector obtained from the normalization of the difference vector $\boldsymbol{r}(R,t_k) = \left[\boldsymbol{p}(R) - \boldsymbol{p}(V_1,t_k)\right]$ (see Fig. 2). For the case at hand, the covariance matrix of the measurements reduces to

$$\boldsymbol{R}_k = \begin{bmatrix} \sigma^2_{v_x}(V_1,t_k) & 0 & 0 \\ 0 & \sigma^2_{v_y}(V_1,t_k) & 0 \\ 0 & 0 & \sigma^2_{\theta_k} \end{bmatrix}. \tag{26}$$

Case 3. The local INS reading clock has ticked.

If an INS reading — which, as mentioned above, is triggered by the local clock every $1/f_{\text{INS}}$ seconds independently of any VANET communication — occurs when the system is not busy in processing a beacon packet, then the algorithm marks such an instant as t_k and carries out a standalone update. To this aim, it performs the same operations discussed above, i.e., eqs. (8), (9), (15), and (16) with a (linear) mapping $\boldsymbol{h}_k(\boldsymbol{x}_k)$ which consists only of the U-vehicle own velocity reading as in (24). The measurements noise covariance matrix simply reduces to

$$\boldsymbol{R}_k = \begin{bmatrix} \sigma^2_{v_x}(V_1,t_k) & 0 \\ 0 & \sigma^2_{v_y}(V_1,t_k) \end{bmatrix}. \tag{27}$$

Case 4. A beacon packet has been received from a vehicle V which is not part of the selected vehicles in the set $\mathcal{V}_{k-1}$.

When the beacon packet comes from a vehicle $V \notin \mathcal{V}_{k-1}$ and $n_{k-1} < n_{\max}$, where $4n_{\max}$ denotes the maximum state size allowed by design, V is included in the current set of selected vehicles, i.e. $\mathcal{V}_k = \mathcal{V}_{k-1}\cup\{V\}$, and the filter state extended (see also Sec. II-D). Otherwise, a decision must be

taken: in fact, if tracking the vehicle V is not more convenient than keep tracking the current selected vehicles, the packet should be discarded; conversely, if tracking the vehicle V instead of at least one of the currently-tracked vehicles is beneficial, V should enter the current set of selected nodes $\mathcal{V}_k$ taking the place of the "worst" vehicle, say V_i, i.e.,

$$\mathcal{V}_k = \{\mathcal{V}_{k-1} \setminus \{V_i\}\} \cup \{V\}. \tag{28}$$

Several strategies can be conceived to establish whether the node can bring advantage to the positioning process or not. We will outline a greedy selection procedure in Sec. IV based on the outcomes of the performance assessment (Sec. III).

In case of replacement, a redefinition of the current state vector is required. To this aim, at first the algorithm performs prediction/update, as in case 3 (based on the U-vehicle velocity reading at t_k), and then the following exchange takes place

$$\boldsymbol{x}_k^{(i)} \leftarrow \left[p_x(V,t_k)\ v_x(V,t_k)\ p_y(V,t_k)\ v_y(V,t_k)\right] \tag{29}$$

that is, the part of the state related to the vehicle V_i is replaced with the new part associated to the vehicle V. Accordingly,

$$\hat{\boldsymbol{x}}_{k|k}^{(i)} \leftarrow \left[\tilde{p}_x(V,t_k)\ \tilde{v}_x(V,t_k)\ \tilde{p}_y(V,t_k)\ \tilde{v}_y(V,t_k)\right] \tag{30}$$

where, as defined above, $\tilde{z}$ represents the estimate of z carried by the received packet. This substitution also reflects on the covariance matrix $\boldsymbol{P}_{k|k}$. To this aim, it is worth noting that such a matrix has the off-diagonal elements which are related to the correlation created among the variables in the filter's state up to that moment. Due to the exchange of eq. (30), a subset of these elements will no longer represent the actual correlation among the new set of variables in the state at t_k. Since the new $\hat{\boldsymbol{x}}_{k|k}$ is built from the estimates sent by V, the new $\boldsymbol{P}_{k|k}$ is constructed by neglecting (setting to zero) the off-diagonal correlation terms corresponding to the vehicle V_i and by replacing the 4×4 block corresponding to V_i with a diagonal block associated to the vehicle V

$$\boldsymbol{P}_{k|k}^{(i)} = \begin{bmatrix} \tilde{\sigma}_x^2(V,t_k) & 0 & 0 & 0 \\ 0 & \tilde{\sigma}_{v_x}^2(V,t_k) & 0 & 0 \\ 0 & 0 & \tilde{\sigma}_y^2(V,t_k) & 0 \\ 0 & 0 & 0 & \tilde{\sigma}_{v_y}^2(V,t_k) \end{bmatrix} \tag{31}$$

being $\tilde{\sigma}_x(V,t_k)$, $\tilde{\sigma}_{v_x}(V,t_k)$, $\tilde{\sigma}_y(V,t_k)$, and $\tilde{\sigma}_{v_y}(V,t_k)$ the standard deviations of the position and velocity along the x and y directions at t_k, respectively, as extracted from the received beacon packet.

D. Operating Modes and Settings

In this section, we describe the operating conditions of our proposed approach and how to properly set the covariance matrices $\boldsymbol{R}_k$ and $\boldsymbol{P}_{0|0}$.

Regarding $\boldsymbol{R}_k$, recall that the measurements to be filtered are the velocity of the U-vehicle, the velocity of tracked vehicles, and the AOA estimates of packets sent by either tracked vehicles or RSUs. The velocity measurements of cooperating vehicles and the corresponding variances are carried by the received packets; the velocity measurement and the associated variance of the U-vehicle are instead available onboard.

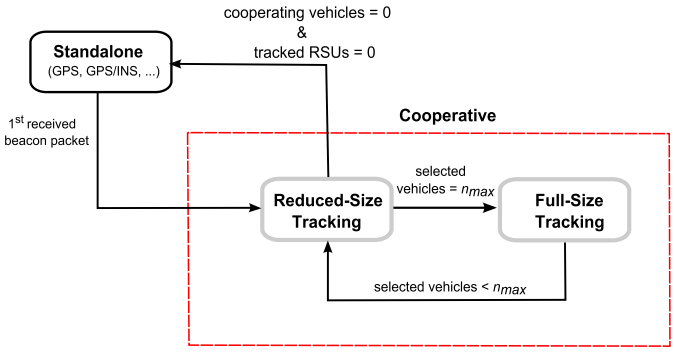


Fig. 3. Algorithm operating modes.

In order to assign a meaningful value to σ_{θ_k}, it should be noticed that, for an array of antennas deployed orthogonally to the longitudinal axis of the vehicle, the Cramér-Rao lower bound (CRLB) for AOA estimation, in additive white Gaussian noise (AWGN) channel, is [46]

$$\mathrm{CRLB}(\theta_k) \propto \frac{1}{\mathrm{SNR}_k \cos^2 \theta_k}. \tag{32}$$

As to SNR_k and θ_k (representing either $\theta(V_i,t_k)$ or $\theta(R,t_k)$ in Fig. 2), they denote the SNR and the angle of arrival at t_k of the impinging packet, respectively. However, in [39] we found that, in fading channels, eq. (32) exhibits a span of several orders of magnitude; hence, to avoid excessive unevennes in the relative weight given to the measurements, we introduced therein a dumping non-linearity, i.e.,

$$\sigma_{\theta_k}^2 = \frac{c}{W \tanh\left(\dfrac{\hat{\mathrm{SNR}}_k \cos^2 \hat{\theta}_k}{W}\right)} \tag{33}$$

where $\hat{\mathrm{SNR}}_k$ is the SNR measured at t_k from the received packet, $\hat{\theta}_k$ is the AOA estimate, and c and W are design parameters. Following the lead of [39], we resort to such a formula for $\sigma_{\theta_k}^2$; the chosen values of c and W are given in Sec. III.

Notice that, differently from the basic version of the EKF considered in the existing literature on CP algorithms for the vehicular context — which assumes synchronous measurements available at the same instant with time-invariant statistics — our algorithm considers asynchronous updates and an adaptive $\boldsymbol{R}_k$ matrix to cope with the intrinsically asynchronous reception of beacon packets and with the (space-) time-dependent nature of measurement variances.

As to $\boldsymbol{P}_{0|0}$, its setting depends on the operating mode the filter starts with. In particular, when the U-vehicle starts up, say at t_0, the filter has to be initialized. In the initial phase the algorithm might work under the *standalone* or the *reduced-size tracking* operating mode (see also Fig. 3); in the former case, localization is based on some standalone technique that exploits only the local information available onboard as described in Sec. I (e.g., GPS, GPS/INS, etc.). Then, as soon as the first beacon packet is received, either from a vehicle or an RSU, a transition occurs from the standalone to the *cooperative* operating mode. In particular, the algorithm enters the so called reduced-size tracking phase.

When the first packet received by the U-vehicle has been sent by a vehicle, the algorithm creates a first instance of the augmented filter by initializing the state vector and the covariance matrix as ($k = 0$ would corresponds to t_0, namely to the case the filter is initialized in the reduced-size tracking operating mode)

$$\hat{\boldsymbol{x}}_{k|k} = \left[\hat{\boldsymbol{x}}^{(1)}\ \hat{\boldsymbol{x}}^{(2)}\right]^T \tag{34}$$

$$\boldsymbol{P}_{k|k} = \text{diag}\left(\boldsymbol{P}^{(1)}_{k|k}, \boldsymbol{P}^{(2)}_{k|k}\right) \tag{35}$$

being $\hat{\boldsymbol{x}}^{(1)}$ the vector of the U-vehicle kinematic information constructed by using the "best" available estimates for position and velocity, and $\boldsymbol{P}^{(1)}_{k|k}$ the 4×4 diagonal block containing the related standard deviations, i.e.,

$$\boldsymbol{P}^{(1)}_{k|k} = \begin{bmatrix} \sigma^2_x(V_1, t_k) & 0 & 0 & 0 \\ 0 & \sigma^2_{v_x}(V_1, t_k) & 0 & 0 \\ 0 & 0 & \sigma^2_y(V_1, t_k) & 0 \\ 0 & 0 & 0 & \sigma^2_{v_y}(V_1, t_k) \end{bmatrix}. \tag{36}$$

Analogously, $\hat{\boldsymbol{x}}^{(2)}$ and the diagonal matrix $\boldsymbol{P}^{(2)}_{k|k}$ are the cooperating vehicle's kinematic data and associated uncertainty, extracted from the received beacon packet. At this time, $\mathcal{V}_k$ includes only the U-vehicle and the cooperating vehicle. If the first received packet has been sent by an RSU, the state will contain only the U-vehicle position and velocity components, i.e., the initialization is $\hat{\boldsymbol{x}}_{k|k} = \hat{\boldsymbol{x}}^{(1)}$ and $\hat{\boldsymbol{P}}_{k|k} = \boldsymbol{P}^{(1)}_{k|k}$. In this case $\mathcal{V}_k$ includes only the U-vehicle.

When the system is in reduced-size tracking mode, every beacon packet received at a generic time t_k from a vehicle not belonging to $\mathcal{V}_{k-1}$ will cause an increase of the filter size (as in Case 4 of Sec. II-C). This will be repeated until the maximum allowed number $n_{\max}$ of selected vehicles is reached. At that point, the algorithm enters the *full-size tracking* mode, keeping to update the filter via the received beacon packets, possibly replacing vehicles in the state according to some criterion (see e.g. Sec. IV).

In case no packet is received for some time from a currently-tracked vehicle, its aging time reaches a maximum value, which likely means that the vehicle is no longer reachable (or active); in that case, the filter state can be reduced by removing the variables related to the inactive node.

If the number of actively cooperating vehicles and RSUs is zero, the algorithm cannot exploit cooperation anymore and, hence, returns back to the standalone mode (see Fig. 3). One might also consider to modify $n_{\max}$ at run-time, based on the fact that, as it will be shown later, the role of vehicles is secondary when even a single active RSU is available, i.e., $n_{\max}$ can be lowered in that case. Similarly, if no active RSU is available, the algorithm could decide to temporarily increase $n_{\max}$, a bit more than in the case of mixed V2I/V2V cooperation, relaxing it as soon as an RSU is detected. This decision depends on the computational budget of the specific implementation, hence will not be further developed here.

E. Other Tracking Approaches

It is now interesting to compare our proposed solution to other methods already existing in literature. It is worth noticing that we were unable to find in the open literature a comprehensive method like the one we are proposing here. Therefore, the comparison we can make here should be limited only to the technique exploited for localization.

In particular, [29], [30] also address the problem of CP using a tracking (KF) approach. However, in all these solutions, a position-only state vector is considered which relies on the following assumed mobility model

$$\boldsymbol{x}_k = \boldsymbol{x}_{k-1} + T_s \boldsymbol{u}_{k-1} + T_s \boldsymbol{w}_{k-1} \tag{37}$$

with

$$\boldsymbol{x}_k = \left[p_x(V_1, t_k) \cdots p_x(V_n, t_k)\ \ p_y(V_1, t_k) \cdots p_y(V_n, t_k)\right]^T \tag{38}$$

$$\begin{aligned} \boldsymbol{u}_{k-1} = [&v_x(V_1, t_{k-1}) \cdots v_x(V_n, t_{k-1}) \\ &\times v_y(V_1, t_{k-1}) \cdots v_y(V_n, t_{k-1})]^T \end{aligned} \tag{39}$$

where $V_i, i = 1, \ldots, n$, denote the IDs of the tracked vehicles at time instant t_{k-1} and T_s is the *synchronous* sampling interval. It is worth noting that, differently from our proposed approach, velocities are external (noisy) inputs for the system. Although such uncertainty will be taken into account through the INS covariance matrix, this formulation highly suffers from the effects of variations in the velocity profile. Such variations are caused by typical driving behaviors, especially in urban environments where the presence of intersections and traffic lights generates a lot of stop&go situations. This drawback can be tackled by filtering the velocity measurements too, as we propose in our solution. By doing so, we obtain an overall increase of the filter robustness (more details will be given in the next section).

Another key difference concerns the nature of radio measurements adopted to compute additional position information (derived from the cooperative vehicular environment) to be filtered. Parker and Valaee [29] and Mohammadabadi and Valaee [30] use RSS and TOA/TDOA techniques, respectively, in order to compute inter-node distances. As mentioned in Sec. I, in [28] it is analytically proved that neither RSS nor TOA/TDOA ranging methods can satisfy the positioning accuracy required for safety applications. In particular, RSS is too sensitive with respect to uncertainties on the path-loss exponent while TOA/TDOA techniques suffer from the lack of an accurate synchronization. As a matter of fact, the order of timing precision of vehicular communication protocols is about 50 μs whereas the synchronization required to achieve ranging errors of a few meters at run-time must be in the order of ns.

Differently from such methods, in this paper we propose a flexible approach where a minimum subset of the kinematic information within received beacon packets is opportunistically filtered together with the related AOA estimates. This brings a great advantage in terms of computational complexity. Indeed, we already noticed that eq. (23) requires the storage and processing of a $4n_{\max} \times 4n_{\max}$ matrix and the

inversion of a 5×5 matrix (in the most demanding Case 1 of Sec. II-C) at each update time.

III. Simulation Model and Results

In this section, we carry out a preliminary simulation analysis to assess the achievable positioning accuracy of our cooperative localization approach. For the whole analysis we focus on a U-vehicle, so as to inspect all the effects related to the different interactions occurring in the analyzed scenarios. We simulate a realistic vehicular environment by modeling several phenomena and non idealities that are found in real urban scenarios. We then show that the proposed approach outperforms a natural competitor that uses additional information, in particular the full set of $n(n-1)$ RSS-based distance estimates among the n tracked vehicles, while our algorithm uses only $n-1$ AOA estimates (in presence of n tracked vehicles). For each analyzed scenario, a Monte Carlo simulation of 1000 trials is conducted.

A. Simulation Model

Following the ETSI standard [47], we assume a carrier frequency $f_c = 5.9$ GHz, a transmit power $P_{T,dB} = 18$ dBm, and a bandwidth $B = 10$ MHz. The f_{RSU} and $f_{vehicle}$ send frequencies are set to 2 Hz, default value in the ETSI specifications. The wireless propagation is modeled according to [48]; in particular, the path loss at distance d from the transmitter is obtained by the *dual slope model*, i.e.,

$$L_{PL,dB}(d) = \begin{cases} L_{F,dB}(d_0) + 10\gamma_1 \log_{10}\left(\dfrac{d}{d_0}\right), & d_0 < d \le d_c \\ L_{F,dB}(d_0) + 10\gamma_2 \log_{10}\left(\dfrac{d}{d_c}\right) & \\ \quad + 10\gamma_1 \log_{10}\left(\dfrac{d_c}{d_0}\right), & d > d_c \end{cases} \tag{40}$$

where d_0 is the *reference distance*, d_c is the *cutoff distance*, $L_{F,dB}(d_0)$ is the signal attenuation in free space (Friis propagation model [49]) at the distance d_0, and γ_1, γ_2 are two attenuation coefficients. According to [48], the values of the parameters are set to $d_0 = 10$ m, $d_c = 80$ m, $\gamma_1 = 1.9$, and $\gamma_2 = 3.8$. Shadowing effects are also taken into account through a lognormal factor with standard deviation 6 dB.

Rayleigh or Rice fast fading is also considered, to model the non-line-of-sight (NLOS) or line-of-sight (LOS) condition of the link, respectively. Links are randomly assigned to the NLOS class with probability 0.5.

The fading term at time $t_{k,r} = t_k + rT_{\text{samp}}$, $r = 0, \ldots, K-1$, where $T_{\text{samp}} < \frac{1}{2B}$ is the sampling period and K the number of considered snapshots for AOA estimation, is expressed as

$$Z(t_{k,r}) = X(t_{k,r}) + jY(t_{k,r}) \sim \mathcal{CN}(\rho\delta_{\ell 1}, 2\sigma_\ell^2) \tag{41}$$

and $\delta_{\ell 1}$ is the Kronecker symbol. As to ℓ, $\ell = 0$ (resp., $\ell = 1$) implies a Rayleigh (resp., Rice) random variable; moreover for the Ricean fading ρ is set such that $10\log_{10}\frac{\rho^2}{2\sigma_1^2} = 6$ dB. Therefore, the incident signal samples $\{s(t_{k,r}), r = 0, \ldots, K-1\}$, associated to the packet received at time t_k can be written as:

$$s(t_{k,r}) = \alpha_{\text{SNR}}(t_k) Z(t_{k,r}) \tag{42}$$

where $\mathrm{E}[|Z(t_{k,r})|^2] = 1$ and $\alpha_{\text{SNR}}(t_k)$ takes into account the lognormal shadowing; the SNR at time t_k is given by

$$\text{SNR}_k = \frac{\mathrm{E}\left[s^*(t_{k,r})\, \boldsymbol{a}^H(\theta_k) s(t_{k,r}) \boldsymbol{a}(\theta_k)\right]}{P_0} = M \frac{\mathrm{E}\left[\alpha_{\text{SNR}}^2(t_k)\right]}{P_0} \tag{43}$$

with $\mathrm{E}\left[\alpha_{\text{SNR}}^2(t_k)\right]$ computed according to the above path loss model and P_0 the noise power, i.e., $P_0 = k_B T_0 B$, k_B being the Boltzmann constant and T_0 the standard noise temperature. As to * and ^{H}, they denote complex conjugate and conjugate transpose, respectively. Finally, $\boldsymbol{a}(\theta_k)$ denotes the steering vector associated with the angle θ_k and an inter-element distance equal to $\lambda/2$ with $\lambda = c/f_c$ and c the speed of light [39]. For AOA estimation we adopt the MUSIC algorithm with $M = 4$ antennas and $K = 20$ snapshots.

To take into account possible packet losses, we accept as valid received packets only those having SNR values greater than 8 dB assuming that, above this value, a correct decoding of all the information will likely occur.

To simulate a realistic GPS error on localization data, the autoregressive model proposed in [39] is adopted, i.e.,

$$\begin{bmatrix} \Delta X(t_k) \\ \Delta Y(t_k) \end{bmatrix} = \phi \begin{bmatrix} \Delta X(t_{k-1}) \\ \Delta Y(t_{k-1}) \end{bmatrix} + \begin{bmatrix} \epsilon_x(t_k) \\ \epsilon_y(t_k) \end{bmatrix} \tag{44}$$

where $\Delta X(t_k)$ and $\Delta Y(t_k)$ are the GPS error components along the x and y axis respectively, $\phi = 0.9$ is the one-lag correlation coefficient, $\epsilon_x(t_k)$, $\epsilon_y(t_k) \sim \mathcal{N}(0, \sigma_\epsilon^2)$ are process noise, and $\sigma_\epsilon^2 = \left(1 - \phi^2\right)\sigma_{\text{GPS}}^2$ where $\sigma_{\text{GPS}} = 4.5$ meters is the assumed GPS error standard deviation for the urban environment. The rationale for using such settings is that GPS data are typically the output of a navigation filter, which thus yields correlated position errors but, at the same time, with a lower standard deviation compared to raw GPS fixes.

For simulation purposes, we assumed a straight stretch of a two-lane road running parallel to the y (or longitudinal) axis of the adopted Cartesian reference system. As concerns the mobility, the Simulation of Urban MObility (SUMO) [50] road traffic simulator (version 0.30.0 with default setting of the parameters) has been used to generate the trajectories of all the involved vehicles. Particularly, for each vehicle and scenario we used one and the same straight line trajectory, resulting in a total path of 45 seconds, over the whole set of Monte Carlo trials. For the sake of illustration, in Fig. 4 we depict one U-vehicle longitudinal velocity profile as obtained from SUMO mobility traces; the transversal (or x) component of the velocity is zero.

INS measurement errors are modeled as independent Gaussian variables with zero mean and standard deviations equal to 10% of the actual velocities. Otherwise stated, the actual standard deviation of GPS-based position estimates is σ_{GPS} while those of velocity estimates are set to 10% of the actual velocity values. Since only estimated values of such standard deviations are typically available, we assume the EKF knows these values up to a 10% error.

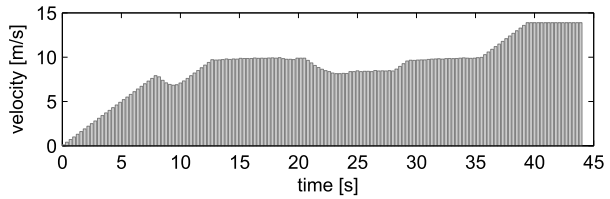


Fig. 4. Longitudinal velocity of the U-vehicle.

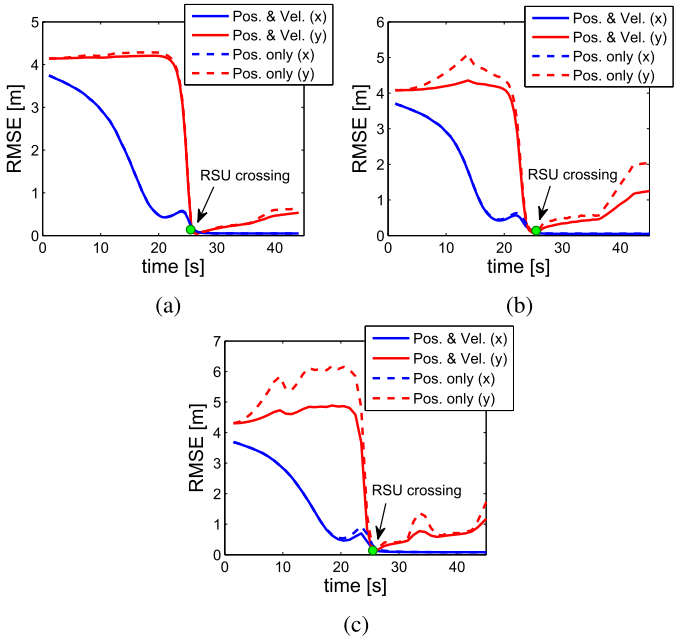


Fig. 5. V2I framework and filters comparison. (a) $f_{INS} = 10$ Hz. (b) $f_{INS} = 4$ Hz. (c) $f_{INS} = 2$ Hz.

Unless differently specified, it is assumed that the CP algorithm initialization is based on GPS estimates. For the nearly constant velocity model described in Sec. II-B, we set $q = 1$, whereas the values adopted for c and W in eq. (33) are 8 and 5000, respectively. As regards the onboard INS measurement rate, we set (unless differently specified) $f_{\text{INS}} = 10$ Hz.

B. Scenario I: V2I-Only Localization

The first analyzed scenario is a minimal situation involving only the U-vehicle and an RSU initially distant 200 m. The U-vehicle moves towards the RSU and crosses it between 20 and 30 s. The root mean square errors (RMSEs) of the vehicle position estimation along the longitudinal (solid red curve) and transverse (solid blue curve) directions during the localization process are represented in Fig. 5a. As it can be seen, the U-vehicle initially knows its position with an error linked to the GPS, but then the effects of filtering come into play and the transverse error begins to decrease. From this behavior it can be deduced that, in the considered scenario that assumes a straight trajectory, as long as the U-vehicle is distant from the RSU, the AOA estimates are more informative for the transverse position estimation while do not provide significant advantage for the longitudinal one. Interestingly, both the RMSE components abruptly drop when the vehicle approaches the RSU, exhibiting values below one meter and thus satisfying the stringent requirements of safety applications. The high level of accuracy is related to the better AOA estimates (notwithstanding values of θ close to 90 degrees) which bring significantly small errors in the state estimates. Then, as the U-vehicle moves far from the RSU, the RMSE curves experience an increasing trend. This behavior is consistent with the fact that AOA estimates become less accurate and, consequently, the filter starts to mainly rely on INS measurements.

In order to show that our choice to filter also the velocities allows the system to better follow the vehicle maneuvering, we make a comparison with the position-only version of the filter. The dashed curves in Fig. 5 represent the RMSEs of the U-vehicle position estimation obtained using such an implementation. It is worth noting that for $f_{INS} = 10$ Hz the solid and dashed curves along the x direction are superimposed. This analysis, conducted assuming different INS reading rates, confirms that using filtered velocities is anyway beneficial. In fact, as figures show, the position-only algorithm highly suffers the effects of reducing the INS reading rate. In particular, one should observe the presence of noticeable gaps between solid and dashed curves, for low rates.

C. Scenario II: V2V-Only Localization

In this second simulation, we assess the performance of the proposed algorithm in a V2V framework, which envisages the presence of cooperating vehicles but no RSUs. In a first setting, we consider the U-vehicle and only one cooperating vehicle V_2, traveling in the opposite direction and initially positioned 200 meters ahead. It is worth noting that this case is really interesting since there are no fixed nodes (RSUs) that can provide their exact locations to the algorithm, and a minimal level of cooperation is considered.

We investigate three different cases: a first case in which both the U-vehicle and V_2 have the same initial position accuracy — in terms of standard deviation of the position error along both x and y directions — of the GPS system ($\sigma_1 = \sigma_2 = 4.5$ m); a second case in which the U-vehicle has worse accuracy ($\sigma_1 = 4.5$ m) than that of V_2 ($\sigma_2 = 0.5$ m); a third case in which the U-vehicle has an initial accuracy ($\sigma_1 = 0.5$ m) higher than that of V_2 ($\sigma_2 = 4.5$ m). Hereafter we assume, without loss of generality, that the position accuracy of cooperating vehicles remains constant during the whole trajectories. To avoid any confusion caused by many curves lying on the same figure, hereafter we will refer to the joint RMSE combining the two error components. Fig. 6 represents the RMSEs of the U-vehicle position estimation for the three cases. The first case shows that, even in case both vehicles start with the GPS localization accuracy, the RMSE obtained after the positioning process (solid curve) is lower. In particular, the U-vehicle reduces its initial RMSE of about 1.6 m at the crossing with V_2, achieving an overall accuracy improvement of approximately 30%.

The localization accuracy can become significantly higher if the tracked vehicle V_2 has a very accurate knowledge of its initial position, as shown by the dashed curve of case 2.

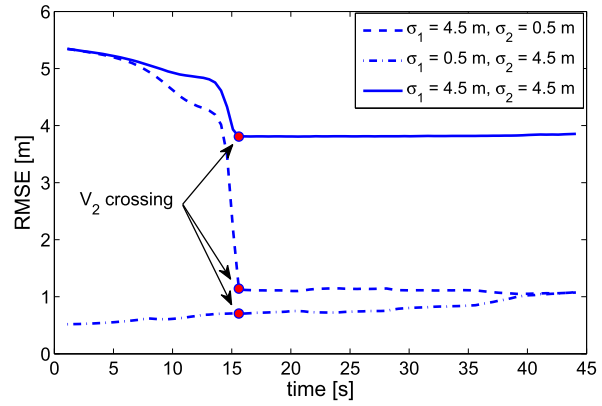


Fig. 6. V2V framework: RMSE of the U-vehicle position estimation.

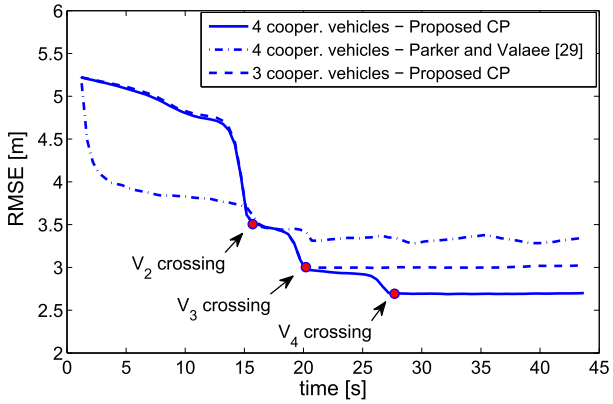


Fig. 7. Different V2V scenarios and comparison against [29].

As a matter of fact, V_2 acts as a mobile anchor that the U-vehicle exploits within our cooperative algorithm, achieving a level of accuracy comparable with that obtained during a V2I localization (Scenario I). Therefore, from these results we can see that the U-vehicle benefits from exploiting nodes with comparable or better initial position accuracy.

At this point, one can reasonably wonder what happens when the tracked neighbor vehicle V_2 has an initial position accuracy worse than that of the U-vehicle. Interestingly, the dash-dot curve of case 3 reveals that the effects of exploiting AOA measurements from a less accurate node are practically negligible. This behavior is consistent with the filtering process which, with an appropriate choice of covariance matrices, severely penalizes both uncertain state estimates and noisy external measurements. As a whole, we can conclude that the proposed CP algorithm is still beneficial also in situations where no fixed anchors (RSUs) are available.

In Fig. 7 we investigate the effects on the U-vehicle localization accuracy when two tracked vehicles, namely V_2 and V_3, are jointly used by the CP algorithm. In particular, we assume that V_2 and V_3 travel in the opposite direction, starting 200 m and 300 m ahead from the U-vehicle, respectively. The dashed curve in Fig. 7 represents the resulting RMSE of the U-vehicle position estimation. We observe that the localization accuracy further improves compared to the case of single vehicle (solid curve in Fig. 6). In particular, by inspecting the RMSE trend we can easily recognize the effects of the two distinct tracked vehicles. Quantitatively, in this example the U-vehicle reduces its initial RMSE of about 2.2 m achieving an accuracy improvement of approximately 40% at the crossing with V_3. From this result, we can highlight that the CP algorithm benefits from using an additional vehicle; however, the additional gain is lower and amounts to about 10%.

To further investigate the positioning accuracy trend, we examine a case in which the set of the currently tracked vehicles includes an additional vehicle V_4, traveling in the opposite direction and starting 400 m ahead from the U-vehicle. The resulting RMSE of the U-vehicle position estimation is represented by the solid curve in Fig. 7. We can observe that the overall position accuracy of the U-vehicle after the localization process is further improved, but the additional gain brought by the fourth vehicle is about 6% at the crossing with V_4. From this analysis, we can conclude that the marginal improvement obtained by exploiting additional nodes has a decreasing trend. This means that three or four cooperating vehicles are already sufficient to obtain almost the maximum gain achievable by exploiting only GPS-localized vehicles.

To complete the analysis, we implemented the EKF-based CP competitor proposed in [29], which shares with our work the most compatible setup among all those described in Sec. I. Since [29] does not foresee the use of RSUs, we perform the comparison in a V2V framework; in particular, we take the same case of $n = 4$ cooperating vehicles discussed above as reference scenario. It is worth remarking that the algorithm in [29] additionally exploits RSS estimates between all cooperating vehicle pairs, i.e., $n(n-1)$ measurements, while we consider only the links between the U-vehicle and selected nodes to estimate $n-1$ AOA.

The dash-dot curve in Fig. 7 represents the RMSE of the U-vehicle position estimation computed according to [29], where a $\sigma_d = 10$ m for the (Gaussian) error of the RSS-based inter-vehicle distance measurements has been assumed. Notice that this assumption is very optimistic for the considered urban environment, since the noise characteristics of RSS-based ranging actually exhibit standard deviations which may exceed 50 meters due to the uncertainty on the path loss exponent and a non-Gaussian error distribution [28]. From the figure it is clear that our approach achieves a better localization accuracy than the algorithm [29] even with one vehicle less.

D. Scenario III: Mixed V2I/V2V Localization

In this third scenario, we analyze a more general case in which both infrastructure (RSU) and vehicle nodes are present. In particular, we consider a very sparse scenario where only three nodes are present: the U-vehicle (V_1), the vehicle V_2, initially positioned 200 meters ahead and traveling in the opposite direction, and an RSU placed on the right-hand side of the street, 350 m ahead from the U-vehicle (at $t = 0$ s; we can imagine, for example, an intersection). We focus on this minimal mixed V2I/V2V scenario to demonstrate the achievable position accuracy in non-favorable conditions; performance can then only improve for more favorable cases.

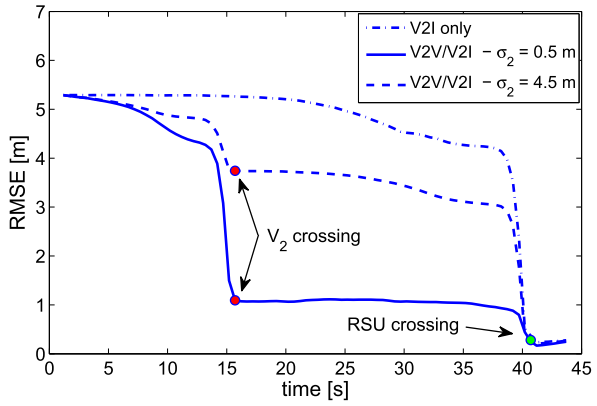


Fig. 8. Mixed V2I/V2V framework: RMSE of the U-vehicle position estimation.

Fig. 8 reports the RMSE of the U-vehicle position estimation and, for comparison, the RMSE for a case where only the RSU cooperates with the U-vehicle (Scenario I, dash-dot curve). The U-vehicle starts with the GPS position accuracy while V_2 has either the same initial position accuracy ($\sigma_2 = 4.5$ m) or a more accurate one ($\sigma_2 = 0.5$ m). It is immediate to observe that the presence of V_2 brings noticeable advantages compared to the V2I-only case. Particularly, the localization accuracy of the U-vehicle increases at the crossing with V_2, despite the considerable distance from the RSU (in agreement with the results of Fig. 6 on the V2V framework). The improvement is about 30% when $\sigma_2 = 4.5$ m and can become significantly higher if an accurate V_2 is considered (as shown by the solid curve). Then, as soon as V_1 approaches the RSU, its accuracy further improves; this should not surprise, since the CP algorithm mainly exploits the AOA measurements obtained from V2I traffic, which is always more informative compared to vehicle traffic due to the known positions of RSUs. This comparative analysis confirms the convenience of a dynamic approach as the one we propose, in which the additional presence of cooperating vehicles can effectively enhance the localization process, especially in low infrastructured contexts.

IV. Tracked Vehicles Selection

On the basis of the lesson learned from the analysis of the simulation results, it is possible to outline a convenient greedy strategy in order to decide possible replacements of vehicles in the filter state when operating in full-size tracking mode. As previously observed, RSU nodes provide the most valuable information for the CP algorithm, thanks to their known positions. Indeed, an RSU can help the U-vehicle to achieve a high level of positioning accuracy regardless of the number and quality of the currently tracked vehicles. In addition, RSUs do not need to be tracked: in the proposed approach their packets are always used to perform a filter update as they arrive, without any limitation since RSU-related variables are never included in filter state. Conversely, each tracked vehicle contributes with 4 variables in the filter state, hence $n_{\max}$ tracked vehicles imply matrix operations of size $4n_{\max} \times 4n_{\max}$. Given the superlinear complexity of such processing, it is convenient to set $n_{\max}$ small, so that a selection procedure is required to decide admission/exclusion from the filter state.

Clearly, when the localization process is mainly based on vehicle nodes, the quality of the information they provide is crucial to obtain adequate performance. On the one hand, this means that the algorithm should prefer nodes whose beacon packets lead to an accurate AOA resolution, measured through a reliability coefficient. Recalling the discussion in Sec. II-D, the latter can be defined as

$$\eta = \hat{\mathrm{SNR}} \cos^2 \hat{\theta} \tag{45}$$

which takes into account both the channel effects and the geometrical conditions affecting the AOA estimate $\hat{\theta}$. On the other hand, from Sec. III it emerged that the ultimate localization performance (RMSE) depends on the accuracy of the tracked vehicles' initial positions, which are used to initialize the corresponding filter state variables. In order to consider both the position accuracy and the AOA measurement quality, we might define the following *goodness* metric

$$G = \frac{\hat{\mathrm{SNR}} \cos^2 \hat{\theta}}{\sqrt{\sigma_x^2 + \sigma_y^2}} = \frac{\eta}{\sigma} \tag{46}$$

being σ_x and σ_y the standard deviations related to the position information along the x and y directions, respectively.

Having defined a candidate metric, we now explain the proposed strategy. The U-vehicle manages a table of attributes related to the current set of tracked vehicles $\mathcal{V}_{k-1} \setminus \{V_1\}$, containing for each *ID* $V_i \in \mathcal{V}_{k-1} \setminus \{V_1\}$ the associated goodness, say G_{V_i}, and the aging time T_{V_i} already described in Sec. II-B. Update of G_{V_i} is performed whenever a beacon packet from vehicle V_i is received, by using the current estimates of SNR and AOA to refresh the associated η, say η_i; conversely, the associated standard deviation, say σ_i, will be updated using the current values from the corresponding matrix block in $\boldsymbol{P}_{k|k}$.

Based on such information, it is possible to devise a greedy selection procedure, aimed at taking the most convenient decision when a beacon packet is received from a vehicle $V \notin \mathcal{V}_{k-1}$. The steps are as follows:

1) G_V is computed from the information associated with the beacon packet, and all G_{V_i}, $V_i \in \mathcal{V}_{k-1}$, are updated in their σ_i factor[4] as described above;
2) G_V is compared against the minimum G_{V_i};
3) if $G_V > \min_{V_i \in \mathcal{V}_{k-1} \setminus \{V_1\}} G_{V_i}$, then V is selected to join the current set of tracked vehicles by replacing a vehicle corresponding to the minimum, as described in Sec. II-C; otherwise, the received beacon packet is discarded.

As mentioned, a vehicle V_i is *inactive* if its T_{V_i} reaches a maximum value: in that case, by setting $G_{V_i} = -\infty$, the greedy procedure given above will automatically replace an inactive vehicle with any new vehicle as soon as available.

[4]The η_i factor will remain updated at the last packet received from V_i, which is typically quite recent; if this is not the case, it means that the aging time will quickly reach the maximum value, so making vehicle V_i eligible for priority replacement.

V. Conclusion

In this work, we have designed a lightweight CP algorithm that is able to suitably adapt to different vehicular contexts. Our solution is based on a variant of the EKF tracking algorithm, with asynchronous updates and dynamic setting of parameters. The algorithm selects a limited set of vehicles according to a chosen strategy, e.g. based on some goodness metric of the data they broadcast; then, the available information — which may include AOA estimates and velocity measurements — is properly fused for the localization process. While most CP algorithms require dense network topologies, our approach can effectively cope with situations involving a very limited number of cooperating nodes. A preliminary analysis of the proposed algorithm has been conducted by means of simulations considering realistic urban environments. The "steady state" analysis of the overall CP algorithm is part of the ongoing research activity. It requires the implementation of the algorithm via a network simulator (e.g., NS3). However, the preliminary results indicate that a considerable accuracy improvement can be achieved even when vehicles with inaccurate position information are exploited, outperforming natural competitors while keeping the system efficient in terms of computation and communication overhead.

References

[1] J. H. L. Hansen, K. Takeda, S. M. Naik, M. M. Trivedi, G. U. Schmidt, and J. Chen, "Signal processing for smart vehicle technologies [from the guest editors]," *IEEE Signal Process. Mag.*, vol. 33, no. 6, pp. 12–13, Nov. 2016.

[2] J. H. L. Hansen *et al.*, "Signal processing for smart vehicle technologies: Part 2," *IEEE Signal Process. Mag.*, vol. 34, no. 2, pp. 18–21, Mar. 2017.

[3] J. Piao, M. Beecroft, and M. McDonald, "Vehicle positioning for improving road safety," *Transp. Rev.*, vol. 30, no. 6, pp. 701–715, Jun. 2010.

[4] S. E. Shladover and S.-K. Tan, "Analysis of vehicle positioning accuracy requirements for communication-based cooperative collision warning," *J. Intell. Transp. Syst.*, vol. 10, no. 3, pp. 131–140, Jan. 2006.

[5] E. D. Kaplan and C. J. Hegarty, *Understanding GPS: Principles and Applications*, 2nd ed. Norwood, MA, USA: Artech House, 2006.

[6] W.-W. Kao, "Integration of GPS and dead-reckoning navigation systems," in *Proc. Vehicle Navigat. Inf. Syst. Conf.*, Dearborn, MI, USA, Oct. 1991, pp. 635–643.

[7] H. Qi and J. B. Moore, "Direct Kalman filtering approach for GPS/INS integration," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 38, no. 2, pp. 687–693, Apr. 2002.

[8] R. Sharaf, A. Noureldin, A. Osman, and N. El-Sheimy, "Online INS/GPS integration with a radial basis function neural network," *IEEE Aerosp. Electron. Syst. Mag.*, vol. 20, no. 3, pp. 8–14, Mar. 2005.

[9] E. S. Abdolkarimi, M. R. Mosavi, A. A. Abedi, and S. Mirzakuchaki, "Optimization of the low-cost INS/GPS navigation system using ANFIS for high speed vehicle application," in *Proc. Signal Process. Intell. Syst. Conf. (SPIS)*, Tehran, Iran, Dec. 2015, pp. 93–98.

[10] H. Liu, S. Nassar, and N. El-Sheimy, "Two-filter smoothing for accurate INS/GPS land-vehicle navigation in urban centers," *IEEE Trans. Veh. Technol.*, vol. 59, no. 9, pp. 4256–4267, Nov. 2010.

[11] A. Coluccia, F. Ricciato, and G. Ricci, "Positioning based on signals of opportunity," *IEEE Commun. Lett.*, vol. 18, no. 2, pp. 356–359, Feb. 2014.

[12] H. Nikookar and P. Oonincx, "An introduction to radio locationing with signals of opportunity," *J. Commun., Navigat., Sens. Services*, vol. 1, no. 1, Jan. 2016, Art. no. 1.

[13] J. J. Morales, P. F. Roysdon, and Z. M. Kassas, "Signals of opportunity aided inertial navigation," in *Proc. Inst. Navigat. (ION) GNSS*, Portland, OR, USA, Sep. 2016, pp. 1492–1501.

[14] C. Yang, T. Nguyen, D. Venable, M. White, and R. Siegel, "Cooperative position location with signals of opportunity," in *Proc. IEEE Nat. Aerosp. Electron. Conf. (NAECON)*, Dayton, OH, USA, Jul. 2009, pp. 18–25.

[15] A. Dammann, S. Sand, and R. Raulefs, "Signals of opportunity in mobile radio positioning," in *Proc. 20th Eur. Signal Process. Conf. (EUSIPCO)*, Bucharest, Romania, Aug. 2012, pp. 549–553.

[16] G. De Angelis, G. Baruffa, and S. Cacopardi, "GNSS/cellular hybrid positioning system for mobile users in urban scenarios," *IEEE Trans. Intell. Transp. Syst.*, vol. 24, no. 1, pp. 313–321, Mar. 2013.

[17] S. Gezici, "A survey on wireless position estimation," *Wireless Pers. Commun.*, vol. 44, no. 3, pp. 263–282, Feb. 2008.

[18] L. Fang *et al.*, "Design of a wireless assisted pedestrian dead reckoning system—The NavMote experience," *IEEE Trans. Instrum. Meas.*, vol. 54, no. 6, pp. 2342–2358, Dec. 2005.

[19] H. Liu *et al.*, "Push the limit of WiFi based localization for smartphones," in *Proc. 18th Annu. Int. Conf. Mobile Comput. Netw.*, Istanbul, Turkey, Aug. 2012, pp. 305–316.

[20] A. Rai, K. K. Chintalapudi, V. N. Padmanabhan, and R. Sen, "Zee: Zero-effort crowdsourcing for indoor localization," in *Proc. 18th Annu. Int. Conf. Mobile Comput. Netw.*, Istanbul, Turkey, Aug. 2012, pp. 293–304.

[21] H. Liu, J. Yang, S. Sidhom, Y. Wang, Y. Chen, and F. Ye, "Accurate WiFi based localization for smartphones using peer assistance," *IEEE Trans. Mobile Comput.*, vol. 13, no. 10, pp. 2199–2214, Oct. 2014.

[22] J. F. Raquet, M. M. Miller, and T. Q. Nguyen, "Issues and approaches for navigation using signals of opportunity," in *Proc. Nat. Tech. Meet. Inst. Navigat.*, San Diego, CA, USA, Jan. 2007, pp. 1073–1080.

[23] M. Leng, W. P. Tay, C. M. S. See, S. G. Razul, and M. Z. Win, "Modified CRLB for cooperative geolocation of two devices using signals of opportunity," *IEEE Trans. Wireless Commun.*, vol. 13, no. 7, pp. 3636–3649, Jul. 2014.

[24] R. Karlsson and F. Gustafsson, "The future of automotive localization algorithms: Available, reliable, and scalable localization: Anywhere and anytime," *IEEE Signal Process. Mag.*, vol. 34, no. 2, pp. 60–69, Mar. 2017.

[25] H. S. Ramos, A. Boukerche, R. W. Pazzi, A. C. Frery, and A. A. F. Loureiro, "Cooperative target tracking in vehicular sensor networks," *IEEE Wireless Commun.*, vol. 19, no. 5, pp. 66–73, Oct. 2012.

[26] E. C. Eze, S. Zhang, and E. Liu, "Vehicular ad hoc networks (VANETs): Current state, challenges, potentials and way forward," in *Proc. 20th IEEE Int. Conf. Autom. Comput. (ICAC)*, Cranfield, U.K., Sep. 2014, pp. 176–181.

[27] A. Boukerche, H. A. B. F. Oliveira, E. F. Nakamura, and A. A. F. Loureiro, "Vehicular ad hoc networks: A new challenge for localization-based systems," *Comput. Commun.*, vol. 31, no. 12, pp. 2838–2849, 2008.

[28] N. Alam and A. G. Dempster, "Cooperative positioning for vehicular networks: Facts and future," *IEEE Trans. Intell. Transp. Syst.*, vol. 14, no. 4, pp. 1708–1717, Dec. 2013.

[29] R. Parker and S. Valaee, "Vehicular node localization using received-signal-strength indicator," *IEEE Trans. Veh. Technol.*, vol. 56, no. 6, pp. 3371–3380, Nov. 2007.

[30] P. H. Mohammadabadi and S. Valaee, "Cooperative node positioning in vehicular networks using inter-node distance measurements," in *Proc. 25th IEEE Annu. Int. Symp. Pers., Indoor, Mobile Radio Commun. (PIMRC)*, Washington, DC, USA, Sep. 2014, pp. 1448–1452.

[31] N. Alam, A. T. Balaei, and A. G. Dempster, "Relative positioning enhancement in VANETs: A tight integration approach," *IEEE Trans. Intell. Transp. Syst.*, vol. 14, no. 1, pp. 47–55, Mar. 2013.

[32] N. Alam, A. Kealy, and A. G. Dempster, "An INS-aided tight integration approach for relative positioning enhancement in VANETs," *IEEE Trans. Intell. Transp. Syst.*, vol. 14, no. 4, pp. 1992–1996, Dec. 2013.

[33] F. de Ponte Müller, E. M. Diaz, B. Kloiber, and T. Strang, "Bayesian cooperative relative vehicle positioning using pseudorange differences," in *Proc. IEEE/ION Position, Location Navigat. Symp. PLANS*, Monterey, CA, USA, May 2014, pp. 434–444.

[34] N. Alam, A. Kealy, and A. G. Dempster, "Cooperative inertial navigation for GNSS-challenged vehicular environments," *IEEE Trans. Intell. Transp. Syst.*, vol. 14, no. 3, pp. 1370–1379, Sep. 2013.

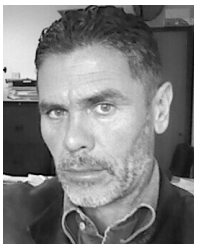
[35] N. Alam, A. T. Balaei, and A. G. Dempster, "An instantaneous lane-level positioning using DSRC carrier frequency offset," *IEEE Trans. Intell. Transp. Syst.*, vol. 13, no. 4, pp. 1566–1575, Dec. 2012.

[36] M. M. Alotaibi, A. Boukerche, and H. Mouftah, "Distributed relative cooperative positioning in vehicular ad-hoc networks," in *Proc. IEEE Global Inf. Infrastruct. Netw. Symp. (GIIS)*, Montreal, QC, Canada, Sep. 2014, pp. 1–8.

[37] A. A. Wahab, A. Khattab, and Y. A. Fahmy, "Two-way TOA with limited dead reckoning for GPS-free vehicle localization using single RSU," in *Proc. 13th Int. Conf. ITS Telecommun. (ITST)*, Tampere, Finland, Nov. 2013, pp. 244–249.

[38] S. Moser, S. Eckert, and F. Slomka, "An approach for the integration of smart antennas in the design and simulation of vehicular ad-hoc networks," in *Proc. 1st Int. Conf. Future Generat. Commun. Technol. (FGCT)*, London, U.K., Dec. 2012, pp. 36–41.
[39] A. Fascista, G. Ciccarese, A. Coluccia, and G. Ricci, "A localization algorithm based on V2I communications and AOA estimation," *IEEE Signal Process. Lett.*, vol. 24, no. 1, pp. 126–130, Jan. 2017.
[40] H. Kloeden, D. Schwarz, E. M. Biebl, and R. H. Rasshofer, "Vehicle localization using cooperative RF-based landmarks," in *Proc. IEEE Intell. Vehicles Symp. (IV)*, Baden-Baden, Germany, Jun. 2011, pp. 387–392.
[41] R. O. Schmidt, "Multiple emitter location and signal parameter estimation," *IEEE Trans. Antennas Propag.*, vol. AP-34, no. 3, pp. 276–280, Mar. 1986.
[42] D. Titterton and J. L. Weston, *Strapdown Inertial Navigation Technology*. London, U.K.: IEE, 2004.
[43] M. S. Grewal, L. R. Weil, and A. P. Andrews, *Global Positioning Systems, Inertial Navigation, and Integration*. Hoboken, NJ, USA: Wiley, 2001.
[44] Y. Bar-Shalom and T. E. Fortmann, *Tracking and Data Association*. San Diego, CA, USA: Academic, 1988.
[45] Y. Bar-Shalom, X. R. Li, and T. Kirubarajan, *Estimation With Application to Tracking Navigation: Theory, Algorithms Software*. Hoboken, NJ, USA: Wiley, 2001.
[46] J. R. Sklar and F. C. Schweppe, "On the angular resolution of multiple targets," *Proc. IEEE*, vol. 52, no. 9, pp. 1044–1045, Sep. 1964.
[47] *European Profile Standard for the Physical and Medium Access Control Layer of Intelligent Transport Systems Operating in the 5 GHz Frequency Band*, document ETSI TS 202 663, Nov. 2009.
[48] *Intelligent Transport Systems (ITS); STDMA Recommended Parameters and Settings for Cooperative ITS; Access Layer Part*, document ETSI TR 102 861, Jan. 2012.
[49] T. S. Rappaport, *Wireless Communications: Principles and Practice*. Englewood Cliffs, NJ, USA: Prentice-Hall, 2001.
[50] D. Krajzewicz, J. Erdmann, M. Behrisch, and L. Bieker, "Recent development and applications of SUMO—Simulation of Urban MObility," *Int. J. Adv. Syst. Meas.*, vol. 5, nos. 3–4, pp. 128–138, Dec. 2012.

Alessio Fascista was born in Lecce, Italy, in 1991. He received the Eng. degree (*summa cum laude*) in computer engineering from University of Salento, Lecce, in 2015, where he is currently pursuing the Ph.D. degree in engineering of complex systems with the Department of Engineering for Innovation. His main research interests are in the field of networking and telecommunications with focus on localization systems for safety critical applications.



Giovanni Ciccarese received the Laurea degree in electronic engineering from Politecnico di Torino, Italy, in 1989. From 1995 to 1996, he was a CNR (the Italian National Research Council) scholarship holder. Joining the Computer Networks Laboratory, Department of Engineering for Innovation, University of Salento, in 1996, he has had significant experiences as a Network Manager and a Researcher in the areas of computer networks. He is currently an Assistant Professor with the Department of Engineering for Innovation, where he is the Head of the Computer Networks Laboratory and teaches the course of network technologies. His research interests are in the field of the design, modeling, and performance evaluation of protocols for wireless networks, with particular emphasis on protocols for vehicular ad hoc networks.

Angelo Coluccia (M'13–SM'16) received the Eng. degree (*summa cum laude*) in telecommunication engineering and Ph.D. degree in information engineering from University of Salento, Lecce, Italy, in 2007 and 2011, respectively. He is currently an Assistant Professor with the Department of Engineering for Innovation, University of Salento, where he teaches the course of telecommunication systems. Since 2008, he has been engaged in research projects on traffic analysis in operational cellular networks, security, and anomaly detection. He has been a Research Fellow with Forschungszentrum Telekommunikation Wien, Vienna, Austria, and a Visiting Scholar with the Institut Supérieur de l'Aéronautique et de l'Espace ISAE-Supaero, Toulouse, France. His research interests are in the area of multi-channel and multi-agent statistical signal processing, including cooperative sensing and estimation in wireless networks, detection, and localization. Relevant application fields are radar, ad-hoc (sensor, overlay, and social) networks, and intelligent transportation systems.

Giuseppe Ricci (M'01–SM'10) was born in Naples, Italy, in 1964. He received the Dr. and Ph.D. degrees in electronic engineering from University of Naples "Federico II" in 1990 and 1994, respectively. Since 1995, he has been with University of Salento first as an Assistant Professor of telecommunications and, since 2002, as a Professor. His research interests are in the field of statistical signal processing with emphasis on radar processing, localization algorithms, and CDMA systems. More precisely, he has focused on high-resolution radar clutter modeling, the detection of radar signals in Gaussian and non-Gaussian disturbance, oil spill detection from SAR data, track-before-detect algorithms fed by space-time radar data, localization in wireless sensor networks, multiuser detection in overlay CDMA systems, and blind multiuser detection. He has held visiting positions with University of Colorado at Boulder, Boulder, CO, USA, in 1997–1998 and in 2001, at Colorado State University, Fort Collins, CO, USA, in 2003, 2005, 2009, and 2011, at Ensica, Toulouse, France, in 2006, and at University of Connecticut, Storrs, CT, USA, in 2008.