IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received February 3, 2019, accepted February 23, 2019, date of publication March 5, 2019, date of current version March 26, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2903219

Adaptive Cooperative Localization Using Relative Position Estimation for Networked Systems With Minimum Number of Communication Links

ALI SAFAEI AND MUHAMMAD NASIRUDDIN MAHYUDDIN, (Senior Member, IEEE)
School of Electrical and Electronic Engineering, Universiti Sains Malaysia, Nibong Tebal 14300, Malaysia

Corresponding author: Muhammad Nasiruddin Mahyuddin (nasiruddin@usm.my)

The work (Ph.D. studies) of A. Safaei was fully supported by the TWAS-USM Postgraduate Fellowship funded by the World Academy of Sciences. This work was supported in part by the USM Research University (RUi) under Grant 1001/PELECT/8014029, in part by the RCMO Bridging Research under Grant 304/PELECT/6316106, and in part by the School of Electrical and Electronic Engineering KPI fund awarded from Universiti Sains Malaysia.

ABSTRACT In this paper, a novel solution for cooperative localization problem involving a network with multiple mobile agents accessing to only one beacon agent is presented. The solution can be applied to a network with a minimum number of communication links (subjecting the network topology to be an undirected *spanning tree*). The objective is to provide each agent in the network with information on its absolute position without using any on-board global positioning sensors such as GPS receivers. The proposed solution uses an adaptive relative position estimation algorithm for each pair of neighboring mobile agents. The proposed estimation algorithm requires each agent to measure the relative inter-agent distance. The local velocity vector is also measured and transmitted to the neighboring agents. The estimation mechanism incorporating a signum function outperforms the recently established relative position estimation algorithms, in terms of positioning and tracking errors. An adaptive cooperative localization (ACL) algorithm is formed by augmenting the relative position estimation in a cooperative observer scheme suitably applicable for accomplishing localization task involving a network of mobile agents. The salient feature of the proposed ACL algorithm is that the communication graph among the agents needs only to have one undirected path between two agents in the network. Such convenience promotes easy practical implementation and lite computation for each agent. The proof of the proposed algorithm is provided using the Lyapunov stability theorem. Three simulation case studies are presented to evaluate the performance of the solution in different scenarios, including the stationary and moving beacon agent as well as the non-cooperatively controlled and cooperatively controlled network of mobile agents. The comparative studies reveal that the ACL algorithm is superior to the recently investigated linear-convex algorithm. The number of communication links required for the localization task to be carried out by the proposed algorithm is minimum, thus promoting a more preferable energy-efficient solution.

INDEX TERMS Relative position estimation, adaptive cooperative localization, multi-agent systems, autonomous mobile robot.

I. INTRODUCTION

Knowing the whereabout of static and dynamic objects with reference to a fixed point has been an interesting on-going discussion and academic debate. The objects of interest (human, landmarks, mobile devices and vehicles) may be located in the sea, air or on the ground. Each of the environments has its own challenges and constraints to be dealt with, specifically. The problem of finding the location of desired objects in a global or local coordinates is named as *localization problem*. Nowadays, availability of more accurate positions for any static or dynamic object, specially in indoor environment is indispensable. One of the easy and cost-effective solutions to this problem is using Geographical Positioning System (GPS). GPS modules are no longer scarce, such that they can be found in all mobile telecommunication devices. These modules receive broadcasted signals by at least four satellites located in the earth atmosphere and provide the

The associate editor coordinating the review of this manuscript and approving it for publication was Lin Wang.

longitude, latitude and altitude of the position in the earth global frame. This data is considered as the *absolute position*. It is shown that the position data provided by the commercial GPS modules in open sky conditions has the mean accuracy of 4.9 meters in radius [1]. The range of this error varies according to the time and location subjecting to ambient conditions such as the weather and hindrance exists in the line-of-sight between the GPS module and the satellites. Such error may not have detrimental impact to the localization task involving large dynamic systems like airplane, ship, car and landmark. For a small dynamical system being less than one meter in dimension, the errors in the GPS reading, drastically affect the localization and consequently the navigation tasks. In addition, the GPS signals are not available inside buildings and also in a jammed environment, where the line-of-sight from the satellites to the GPS module is not available. There are numerous solutions proposed in the literature dealing with the aforementioned issues and providing more accurate tools for localization problem. These solutions can be categorized as follows.

- Methods which are trying to improve the accuracy of localization with GPS;
  - *Method-1*: Improving the GPS data accuracy by correction signals received from a node with exact known position. This category contains Differential GPS (DGPS) and Real-Time Kinematics (RTK). Although these methods can provide the accuracy within some centimeters, the high expenses to support the enabling infrastructure prove the methods to be major hurdles in wide installations offered to public customers [2].
  - *Method-2*: Providing GPS modules with the actual information of orbiting satellites in the earth atmosphere. This technique is named as Assisted-GPS (A-GPS) and is available now in smart phones with accuracy of about 10 meters [3].
- Methods without any GPS module on the nodes (or targets) undergone localization process;
  - *Method-3*: By integrating the inertial sensor measurements (so-called *dead-reckoning*), the information about the position and orientation of a mobile target can be obtained with reference to the initial states. Inertial sensors (mostly named as Inertial Measurement Unit or IMU in short) are the sensors for measuring the kinematics parameters of a mobile system [4]. However, errors in the measurement will accumulate and the estimates will drift after a definite time [4]–[7]. There are several filtering methods such as extended Kalman filter (EKF) and particle filter which are proposed to deal with the error drift. However, the aforementioned estimation algorithm relies on the inclusion of white noise signals in the measured data (attributed to the Persistent Excitation (PE) condition) in order to achieve an accurate localization procedure [7]. In this regard, data from a global coordinate frame or some geometric constraints can be utilized so as to have better localization performance. Moreover, incorporating some data available in a network including several agents can be useful. In other words, if each agent only relies on propagating its equation of motion using self-motion measurements, the state estimation error drifts due to the noise with a standard deviation that grows in time without bound. To reduce the growth rate of this estimation error, a *cooperative localization* strategy can be employed [7].
  - *Method-4*: Defining the target location using received data from several fixed or mobile nodes (usually named as *beacon* nodes) in a network. WiFi and Cellular positioning methods belong to this group [3]. In addition, the localization algorithms proposed in the literature for wireless sensor networks (WSNs) are considered in this category [9]–[11], [13]. Here, the beacon nodes are located on pre-defined positions or equipped with the GPS modules (hybrid algorithms). The accuracy of available technologies ranges from 20 to 30 meters for WiFi positioning and from 30 to 60 meters for cellular positioning [3], [8].

While a substantial advancement in the work to solve the localization problem in all of the above categories is evident, cooperative algorithms have been receiving a focal attention, recently. The cooperative paradigm is adopted mainly due to the emerging technologies in connectivity and the trending in embedded control systems in the directions toward Industrial Revolution 4.0. In addition, these solutions have received more attentions in different types of applications including autonomous mobile robots. Collaborative or cooperative localization using multiple mobile agents provides several potential advantages over using single mobile agent, including increased localization accuracy and coverage areas, robustness, and flexibility in case of limited sensing ranges and possible measurement failures due to severe environments [19]. For a network of mobile agents, there are several localization solutions proposed in the literature, which are reviewed with details in [12]. According to that survey, cooperative localization algorithms for the network of mobile agents are divided into five categories as grid-based algorithms, probability distribution solutions, timing-based localization, constrained localization algorithms and the solutions for underwater sensor networks [12]. Among them, the algorithms based on the probability distribution are used in many localization cases. Particularly, the Bay's rule is utilized to provide the likelihood distribution for agents position in the network [41]. These algorithms include EKF and particle filters at each agent in the network. The EKF filters can be used in a network with known dynamic and measurement models for each agent in the network. By applying the EKF filters, likelihood distribution of the agents positions have been approximated by a Gaussian estimation method. This leads to large localization error if the exact distribution for the agents position is not a Gaussian one [41]. On the other hand, particle filters or sequential Monte Carlo methods can be implemented for non-Gaussian systems. In this regard, they are employed in many real practical localization problems. Despite multitude of benefits which the particle filter can offer such as flexibility, simple implementation and no requirement for huge memory to store the past data; these solutions are very time consuming due to several samples needed to produce the likelihood distribution of the agents position [41]. Besides the categories mentioned in above, all of the solutions proposed in the category of cooperative localization algorithms for a network of mobile agents, can be identified with the following properties;

- Ranged-based or range-free; regarding the availability of inter-agent measurements (including distance and bearing) in the network [13].
- Anchor-based or anchor-free; regarding the density of nodes with pre-known location information (beacons or anchors) in the network [14].
- Static or mobile; regarding the mobility of beacon nodes in the network [15], [16].
- Distributed or centralized; depending on whether a central processor exists or not, localization schemes can be divided into centralized schemes and distributed schemes [17], [18].

II. RELATED WORKS

All the available solutions for cooperative localization problem are seeking to reach more accurate information about the position of either each agent in the network (i.e. self-localization) or any other fixed or mobile target in both outdoor and indoor environments. The initial investigation on the cooperative localization is presented in [21], where the *leap-frogging* motion pattern is suggested for a team of mobile agents. The main disadvantage of that algorithm is that, only one mobile agent or at-most a portion of them is allowed to navigate in the environment at each time step, leading to longer completion time of the navigation-localization mission [20]. The leap-frogging technique is further improved by other researchers as in [22]–[24]. All of those solutions are in category of centralized cooperative localization algorithms. Due to the high computational costs of the centralized algorithms, decentralized (or distributed) cooperative localization algorithms are developed so as to reduce the computational cost [20]. One of the first distributed solution for the cooperative localization problem in a team of mobile robots is presented in [25] and [26]. The algorithm is further improved in [27] and huge enhancement of the computational cost has been achieved over the previously centralized solutions. The main drawback is, however, the mobile agent dynamic should be known as a set of *a priori* information. There are methods in the literature which deal with the probabilistic localization in the multi-agent systems. These methods rely less on the agents dynamics; instead they are working on the probability distribution generated based on the previous location of the mobile agents and the inter-agent relative observations. One of the most well-known probabilistic localization techniques, is the Monte Carlo decentralized cooperative localization algorithm, which is presented in [28]–[30]. Furthermore, novel sampling and clustering techniques are proposed in [31] and [32] so as to reduce the set of the particles as well as the computational cost of the Monte Carlo localization solutions. The problem of cross-correlation between the local position estimations at each of the mobile agents is still existed in the above solutions, leading to major unknown error in the localization task [20].

For single-integrator agents, Oh and Ahn [33] have proposed a cooperative position estimation law. It is claimed that the estimated values converge to the actual positions of the agents if the communication graph for the network is uniformly connected. Moreover, the agents in the network actively controls their position using the formation control law utilizing the estimated position. Formation control, which is one of the most actively studied topics within the realm of multi-agent systems, generally aims to drive multiple agents to achieve prescribed inter-agent differences in the state-space [34], [54], [55]. In [7], two distributed EKF algorithms are proposed for position estimation in a team of wheeled mobile robots (WMRs). Similarly, Lu *et al.* [36] proposed a cooperative localization among a team of three WMRs using EKF. In [35], a network of multiple quadrotors is used to track a moving target. Each quadrotor is equipped with a cooperative EKF to estimate the position of the target. Besides, an optimization problem is solved to define the optimal path that quadrotors should move in the 3D space to maximize both the accuracy of their own position estimation and that of the mobile target. Since the cooperative EKF is used in the mentioned solutions, the localization algorithm is coupled with the dynamics of the agents as well as the controller signals. Hence, the performance of the controller affects the cooperative localization algorithm. Algorithm proposed in [5], deals with the problem of mutual localization (or self-localization) and formation control by using vision-based measurements. In general, vision-based sensors give projective measurements that do not contain distance information. As a consequence, it is possible to obtain only bearing information between two agents. In some cases, these measurements can be augmented with distance information by using an additional depth sensor or visible structures with known dimensions. Chai *et al.* [19] aimed to solve the cooperative localization problem for a group of mobile agents with respect to a single landmark. Toward this goal, that paper developed a cooperative estimation scheme for each mobile agent to locate itself, i.e., estimating the relative coordinates of each agent with respect to a stationary landmark. However, the estimation scheme is not applicable for a moving landmark (or beacon). In that paper, every agent is equipped with on-board interoceptive sensors for the measurement of its own absolute velocity; and exteroceptive sensors for the measurement of distances to its nearby agents and the change in rates of the distances. Proposed algorithm in [37], tried to localize three flying quadrotors using a mobile anchor located on a WMR. The localization is not performed in a global coordinate system. Instead, the performance of the proposed method is investigated in an outdoor area covered by eight cameras. Actually, there is local inertial coordinate system and the quadrotors' positions are estimated with regards to the origin of that frame. In [38], the distributed localization problem alongside with the mobile target tracking problem is solved for a network of agents with nonlinear dynamics. There, the decentralized cooperative position estimation is performed using the nonparametric belief propagation algorithm. The problem addressed in [6] is to localize a GPS-denied unmanned aerial vehicle (UAV) (Agent B) with the assistance of a nearby GPS-enabled UAV (Agent A). Agent A broadcasts its global coordinates at discrete instants in time. Both agents move around arbitrarily in 3D space. There is a need for measurement of angle of arrival for the signals communicated between the two agents. While this type of measurement can be available easily for the stationary agents, it is hard to synchronize the measurements of the angle of arrival and the relative distance for the mobile agents such as quadrotors in practical applications [39].

In one of the most recent works on the cooperative localization, an integrated solution for relative localization and leader-follower formation control in a team of quadrotors has been proposed in [39]. The work made an assumption that each quadrotor has on-board sensors to measure the local velocity (using IMU module) and the inter-agent relative distance (using ultra-wide-band (UWB) module). The objective was to estimate the relative position of the agents in a 2D environment. There was not any aim to estimate the absolute position of the agents and localize them in a global frame. In this regard, the communication graph between the agents needs to have at-least two paths from one agent in the network to reach another agent, which is restrictive and often hard to be practically achieved.

Recently, a distributed algorithm is suggested for cooperative localization within a network of mobile dynamic systems, where the network includes at least one beacon agent and at least three listening agents [40], [41]. The solution uses the popular triangulation method among four neighboring agents in the network. Due to the reliance on the method of inclusion test (at-least $\mathbb{R}^2$ enveloping space), there must be three or more communication links to other agents locally at each agent so as to have the cooperative localization algorithm into account. This imposes a restrictive requirement for practical implementation of the algorithm, where the least possible communication links is more desirable. In addition, the cooperative localization solution proposed in [40] and [41] can not be implemented in a network of less than four mobile agents.

Here in our proposed ACL algorithm, first an adaptive estimation algorithm is proposed for estimating the relative position of two neighboring agents using the measurements on local velocity and the inter-agent relative distance. Then, a cooperative observer is presented for estimating the absolute positions of every agents in the network using the estimated relative positions between the neighboring agents and the absolute position data available at the only beacon agent of the network. The ACL algorithm can be implemented within a network including any number of mobile agents with the minimal communication links among the agents. The algorithm requires the network to be an undirected spanning tree, which provides the minimum possible number of communication links in the network so as to have it to be connected. Moreover, the proposed solution is completely decoupled from the dynamics of agents as well as the control algorithm used at each agent in the network. Hence, it can be applied to any team of nonlinear moving agents with completely unknown dynamics and unknown bounded disturbances. In this regard, the main contributions of the proposed cooperative localization solution can be described as follows;

- the proposed ACL algorithm can provide the absolute positions of all agents using only one beacon agent in a network consisting more than one mobile agent (i.e. one beacon agent plus one listening agent is the minimum requirement);
- the requirement on the communication graph between the agents is minimal and it needs only one undirected path between two adjacent agents, offering practical convenience.

In the rest of the manuscript, first the adaptive relative position estimating algorithm is proposed for two mobile agents with non-zero relative velocity in Section III. Then, a cooperative observer is presented in Section IV in order to use the provided information of the relative position estimation for estimating the absolute positions of all the agents in the network by the help of only one beacon agent. The convergence analyses for the methods are provided using Lyapunov stability theorem. The combination of two methods mentioned above, forms the ACL algorithm for absolute localization in a network of mobile agents. The ACL algorithm is presented in Section V. In Section VI, three simulation case studies including two comparative analyses, are exemplified to observe the performance of the proposed ACL algorithm. These case studies confirm the incorporation of the proposed solution with the non-cooperatively controlled as well as the cooperatively controlled network of mobile agents.

III. ADAPTIVE RELATIVE POSITION ESTIMATION

Definition 1: Consider two moving agents with unknown dynamics establish a communication link between each other, through which the relative distance and the relative velocity of one to another can be measured. In this regard, we define $d_r \in \mathbb{R}^+$ as the relative distance and $V_r \in \mathbb{R}^{n\times 1}$ as the relative velocity. The kinematics of relative motion between the two agents can be presented as follows

$$\dot{P}_r = V_r, \tag{1}$$

where $P_r \in \mathbb{R}^{n\times 1}$ is the unknown unmeasurable relative position between the agents. In this regard, one can observe

$$d_r^2 = P_r^T P_r. \tag{2}$$

Here, $n > 0$ is the number of position dimensions in the environment where the agents are located. Obviously, we have $n \in \{2, 3\}$.

Proposition 1: The objective of the adaptive relative position estimating algorithm is to estimate the relative position between the two agents defined in *Definition 1*, such that the relative position estimation error, i.e.

$$\epsilon = P_r - \hat{P}_r \tag{3}$$

as well as the corresponding distance estimation error defined by

$$e = d_r^2 - \hat{P}_r^T \hat{P}_r, \tag{4}$$

converge to a small set around zero as $t \to \infty$, where $\hat{P}_r \in \mathbb{R}^{n\times 1}$ is the estimated relative position between the two agents and $t > 0$ is the symbol representing time.

Assumption 1: It is assumed that the agents defined in *Definition 1* move in a way that their relative position and velocity are always non-zero and bounded. In this sense, we have $V_m^2 \le V_r^T V_r \le V_M^2$ and $P_m^2 \le P_r^T P_r \le P_M^2$ for $t \in [0, \infty\}$ and $V_M, V_m, P_M, P_m \in \mathbb{R}^+$. Consequently, one can observe $D_m \le d_r \le D_M$ and $D_{dm} \le \dot{d}_r \le D_{dM}$, where $D_M, D_m, D_{dM}, D_{dm} \in \mathbb{R}^+$.

Assumption 2: The orientation of the local frames at the two agents defined in *Definition 1* are consistent [39]. This can be achieved by having access to the orientation of the earth magnetic field and also measuring the Euler angles of the local frames using special IMU modules [42].

Assumption 3: The initial relative position (i.e $P_r(0)$ at $t = 0$) between the agents defined in *Definition 1* is available while they are stationary. According to [39], this can be achieved during an *initialization process* for the two static agents. The algorithm proposed in [43] can be considered as a solution for the initialization process.

Theorem 1: Providing *Assumption 1*, *Assumption 2* and *Assumption 3*, if one can estimate the relative position between the two agents defined in *Definition 1*, by

$$\dot{\hat{P}}_r = [1 + \alpha\ sgn(eg)]V_r, \tag{5}$$

where

$$g = V_r^T \hat{P}_r, \tag{6}$$

$sgn(.)$ is the signum function and $\alpha \in (0, 1]$; then the objectives presented in *Proposition 1* will be achieved.

Proof: Let us define

$$U_1 = \frac{1}{4}e^2. \tag{7}$$

By taking the time-derivative, we have

$$\dot{U}_1 = e(d_r \dot{d}_r - \dot{\hat{P}}_r^T \hat{P}_r). \tag{8}$$

Besides, using the time-derivative of (2), one can reach to

$$d_r \dot{d}_r = V_r^T P_r. \tag{9}$$

By replacing (9) and (5) in (8), we lead to

$$\dot{U}_1 = e[V_r^T P_r - V_r^T \hat{P}_r - \alpha\ sgn(eg) V_r^T \hat{P}_r]. \tag{10}$$

Then, by using (6) and a little rearranging, one can have

$$\dot{U}_1 = e\delta - \alpha|e||g|, \tag{11}$$

where $\delta = V_r^T P_r - V_r^T \hat{P}_r$ and $|.|$ is the symbol for representing the absolute values. Recalling *Assumption 3* and by considering (1) and (5), one can have (for $t > 0$)

$$P_r = P_r(0) + V_r t \tag{12}$$

and

$$\hat{P}_r = P_r(0) + [1 + \alpha\ sgn(eg)]V_r t. \tag{13}$$

Utilizing (12) and (13), δ can be written as

$$\delta = V_r^T P_r(0) + V_r^T V_r t - V_r^T P_r(0) - [1 + \alpha\ sgn(eg)] V_r^T V_r t, \tag{14}$$

and consequently, we reach to

$$\delta = -\alpha\ sgn(eg) V_r^T V_r t. \tag{15}$$

Hence, the first term in the right-hand side of equation in (11) is written as follows

$$e\delta = -\alpha\ sgn(g)|e| V_r^T V_r t. \tag{16}$$

In this equation, we have

$$g = V_r^T P_r(0) + [1 + \alpha\ sgn(eg)] V_r^T V_r t. \tag{17}$$

It can be easily shown that the second term in g is positive as long as $0 < \alpha \le 1$. Thus, by assuming $P(0) = 0$ without any loss of generality, one can say that $g > 0$ and consequently $sgn(g) > 0$. By applying this in (16) and rephrasing (11), one can reach to

$$\dot{U}_1 = -\alpha\ sgn(g)|e| V_r^T V_r t - \alpha|e||g| \le 0. \tag{18}$$

Then, since $U_1 > 0$ and $\dot{U}_1 \le 0$, the value of e converges to zero asymptotically, based on Lyapunov stability theorem. Since d_r is bounded based on *Assumption 1*, and by considering the definition of e in (4), we have

$$P_{hm}^2 \le \hat{P}_r^T \hat{P}_r \le P_{hM}^2, \tag{19}$$

where $P_{hm}^2 = D_m^2$, $P_{hM}^2 = D_M^2$ and $P_{hm}, P_{hM} \in \mathbb{R}^+$. This shows that $\hat{P}_r$ is bounded. Consequently $\epsilon = P_r - \hat{P}_r$ is bounded, since P_r is assumed to be bounded per *Assumption 1*. Thus for $E_M \in \mathbb{R}^+$, one can define

$$\epsilon^T \epsilon \le E_M. \tag{20}$$

In addition, by utilizing (12) and (13), the value of ϵ can be represented as follows

$$\begin{aligned}\epsilon &= P_r(0) + V_r t - P_r(0) - V_r t - \alpha\ sgn(eg) V_r t \\ &= -\alpha\ sgn(eg) V_r t.\end{aligned} \tag{21}$$

Besides, we have

$$\dot{\epsilon} = V_r - [1 + \alpha\ sgn(eg)]V_r = -\alpha\ sgn(eg) V_r = \frac{1}{t}\epsilon. \tag{22}$$

As the second part of the proof, let us define

$$U_2 = \frac{1}{2}\epsilon^T\epsilon + U_1. \tag{23}$$

Then, we would have

$$\dot{U}_2 = \dot{\epsilon}^T\epsilon + \dot{U}_1 \tag{24}$$

By replacing (22), we lead to

$$\dot{U}_2 = \frac{1}{t}\epsilon^T\epsilon + \dot{U}_1. \tag{25}$$

Then, by recalling (18) and (20), we reach to

$$\dot{U}_2 \leq -H + \eta, \tag{26}$$

where

$$H = \alpha\ sgn(g)|e|V_r^T V_r t + \alpha|e||g| \geq 0 \tag{27}$$

and

$$\eta = E_M. \tag{28}$$

Based on LaSalle-Yoshizawa theorem [53], since η is a positive constant value and it is shown previously that e is converging to zero asymptotically, one can say that ϵ is uniformly ultimately bounded and converges to a bounded set around the origin. Then, the objective presented in *Proposition 1* is satisfied and the proof is completed. ■

Remark 1: The adaptive law proposed in (5) can be used for estimating the relative position between each pair of moving agents that have a joint communication link. An intuitive rival for this algorithm is $\dot{\hat{P}}_r = V_r$ [40]. In addition, a novel estimator is presented in [39] to have more suitable estimation of the relative positioning among the moving agents. A comparative study is presented in Section V to show that the proposed algorithm in (5) outperforms the other two methods. Moreover, This algorithm can be utilized as the basis for the cooperative localization within a team including several pair of mobile agents.

IV. COOPERATIVE OBSERVER FOR ESTIMATING THE ABSOLUTE POSITION

Having the relative position between each pair of mobile agents be estimated by (5), a cooperative observer is proposed in this section for estimating the absolute position of the mobile agents in a network.

Definition 2: Consider a network consisting of N heterogeneous agents. Let $\mathcal{G}(\digamma, \mathcal{E}, \mathcal{A})$ be a graph with the set of N *nodes* $\digamma = (v_1, v_2, ..., v_N)$, a set of *edges* $\mathcal{E} = (e_{ij}) \in \mathbb{R}^{N\times N}$ and associated adjacency matrix $\mathcal{A} = (a_{ij}) \in \mathbb{R}^{N\times N}$. An edge e_{ij} in $\mathcal{G}$ is a link between a pair of nodes (v_i, v_j), representing the flow of information from v_j to v_i. The e_{ij} is in existence if and only if $a_{ij} > 0$. The graph is undirected, i.e. the e_{ij} and e_{ji} in $\mathcal{G}$ are considered to be the same. We name v_i and v_j as neighbors, if $e_{ij} \in \mathcal{E}$. The communication graph is considered to be connected, meaning that there is a path between each pair of agents in the network. The in-degree matrix is defined as $\mathcal{D} = diag(d_1, d_2, ..., d_N) \in \mathbb{R}^{N\times N}$, where each d_i is the input degree to each node, i.e. $d_i = \Sigma_{j=1}^N a_{ij}$. Hence, we can define Laplacian matrix $\mathcal{L}$ as below [54], [55]

$$\mathcal{L} = \mathcal{D} - \mathcal{A}. \tag{29}$$

Furthermore, by considering a beacon agent with known absolute position, one can define the *beacon pinning gain matrix* as follows

$$\mathcal{B} = diag(b_1, b_2, ..., b_N) \in \mathbb{R}^{N\times N}, \tag{30}$$

in which b_i indicates the existence of a communication link between the beacon agent and the ith agent in the network [54], [55]. Then, we denote

$$\mathcal{H} = \mathcal{L} + \mathcal{B}. \tag{31}$$

Assumption 4: There is at least one communication link between one of the agents and the beacon. In other words, at least one of the diagonal elements in $\mathcal{B}$ is non-zero.

Proposition 2: For the connected agent i to the beacon agent ($i \in [1, N]$), the position $p^i \in \mathbb{R}^{n\times 1}$ can be presented as [51]

$$p^i = p^b + \delta_{ib}, \tag{32}$$

where $p^b \in \mathbb{R}^{n\times 1}$ is the position for the beacon agent and $\delta_{ib} \in \mathbb{R}^{n\times 1}$ is the relative position between the connected agent and the beacon agent. It is assumed that δ_{ib} is available locally at agent i connected to the beacon.

Proposition 3: For the unconnected agent i to the beacon ($i \in [1, N]$), the position p^i is represented as follows [51]

$$p^i = p^j + \delta_{ij} \tag{33}$$

where $\delta_{ij} \in \mathbb{R}^{n\times 1}$ is the relative position between the agent i and the neighboring agent $j \in [1, N]$. It is assumed that the values of δ_{ij} are available at agent i, if there is a communication link between the agents i and j.

Proposition 4: Suppose that all of the agents in the network are following the conditions presented in *Assumption 1*, *Assumption 2* and *Assumption 3*; then, we can use $\hat{P}_r$ estimated by (5) in *Theorem 1* to compute the values of δ_{ij} and δ_{ib} as the relative position (not relative distance) between the neighboring agents, required in (32) and (33).

Proposition 5: In general, the relative difference between two vectors is identical in all of the *orthogonal* local or global coordinates frames. Hence, the relative difference between the agents position in a local frame $\mathbb{P}_i$ at agent i can be presented as follows

$$\delta_{ij} = [\delta_{ij}]_{\mathbb{P}_i} = [\Delta_i^{\mathbb{P}_i} - \Delta_j^{\mathbb{P}_j}]_{\mathbb{P}_i}, \tag{34}$$

where $\Delta_i^{\mathbb{P}_i} \in \mathbb{R}^{n\times 1}$ for $i \in [1, N]$ is the vector for position of agent i in the local frame $\mathbb{P}_i$. In addition, $[.]_{\mathbb{P}_i}$ is the symbol for presenting the relative difference in the local frame $\mathbb{P}_i$. Since the local frame $\mathbb{P}_i$ is fixed to agent i, the time-derivative of $\Delta_i^{\mathbb{P}_i}$ in $\mathbb{P}_i$ is zero, i.e.

$$[\dot{\Delta}_i^{\mathbb{P}_i}]_{\mathbb{P}_i} = 0. \tag{35}$$

Similarly, we can define

$$\delta_{ib} = [\delta_{ib}]_{\mathbb{P}_i} = [\Delta_i^{\mathbb{P}_i} - \Delta_b^{\mathbb{P}_b}]_{\mathbb{P}_i}, \tag{36}$$

where $\Delta_b^{\mathbb{P}_b} \in \mathbb{R}^{n\times 1}$ is the position of the beacon agent at the local frame $\mathbb{P}_b$ fixed to it.

Definition 3: According to *Proposition 2* and *Proposition 3*, a consensus error can be defined for observing the position of agent i as follows

$$\tau^i = \sum_{j=1}^{N} a_{ij}[\hat{p}^i - (\hat{p}^j + \delta_{ij})] + b_i[\hat{p}^i - (p^b + \delta_{ib})]. \tag{37}$$

By recalling (34) and defining

$$\hat{s}^i = \hat{p}^i - \Delta_i^{\mathbb{P}_i},$$
$$\hat{s}^j = \hat{p}^j - \Delta_j^{\mathbb{P}_j}, \tag{38}$$

the localization consensus error can be represented in a lumped format as follows

$$\tau = (\mathcal{H} \otimes I_n)\hat{s} - (\mathcal{B} \otimes s^b)1, \tag{39}$$

where $\hat{s} = [\hat{s}^1; \hat{s}^2; \ldots; \hat{s}^N]$, $s^b = p^b - \Delta_b^{\mathbb{P}_b}$ and $1 \in \mathbb{R}^{Nn\times 1}$ is a vector with one for all the elements.

Proposition 6: If the consensus error defined in (39) converges to zero, one can say that the distributed observation objective is achieved and $\hat{p}^i$ reaches to p^i for all $i \in [1, N]$.

Definition 4: We define a function $\mathcal{M}(.)$ for generating a diagonal matrix $M \in \mathbb{R}^{n\times n}$ with zero off-diagonal elements, by the elements of a vector $l \in \mathbb{R}^{n\times 1}$ as follows

$$M = \mathcal{M}(l) = \mathcal{M}_l, \tag{40}$$

where $M[j_0, j_0] = l[j_0]$ for $j_0 \in [1, n]$.

Theorem 2: If one uses the following equation as the rate for observing the position of agent i,

$$\dot{\hat{p}}^i = -\lambda\tau^i - [\mathcal{M}(sgn\{\sum_{j=1}^{N}(\mathcal{H}(i,j)\tau^j)\}\ M^b] \tag{41}$$

where $\lambda > 0$ is a scalar, $\mathcal{M}(sgn\{\Sigma_{j=1}^N(\mathcal{H}(i,j)\tau^j)\}) \in \mathbb{R}^{n\times n}$ is a diagonal matrix whose elements on the main diameter are the sign of elements in $\Sigma_{j=1}^N(\mathcal{H}(i,j)\tau^j) \in \mathbb{R}^{n\times 1}$ (refer to *Definition 4*) and $M^b \in \mathbb{R}^{n\times 1}$ includes the maximum absolute values for the elements of $\dot{s}^b$; then the *Proposition 6* can be achieved. *Proof:* . Considering the following Lyapunov function

$$U_3 = \frac{1}{2}\tau^T\tau, \tag{42}$$

one can have

$$\dot{U}_3 = \tau^T[(\mathcal{H} \otimes I_n)\dot{\hat{s}} - (\mathcal{B} \otimes \dot{s}^b)1]. \tag{43}$$

Since the summation of all elements in each row of the Laplacian matrix is zero [54], [55], we can say that,

$$(\mathcal{L} \otimes \dot{s}^b)1 = 0. \tag{44}$$

Hence, (43) can be represented as

$$\dot{U}_3 = \tau^T(\mathcal{H} \otimes I_n)\dot{\hat{s}} - \tau^T(\mathcal{H} \otimes \dot{s}^b)1. \tag{45}$$

Considering $\dot{\hat{s}} = -\lambda\tau + \hat{s}_1$, we have

$$\dot{U}_3 = -\lambda\tau^T(\mathcal{H} \otimes I_n)\tau + \tau^T(\mathcal{H} \otimes I_n)\hat{s}_1 - \tau^T(\mathcal{H} \otimes \dot{s}^b)1. \tag{46}$$

Recalling *Definition 2* and *Assumption 4*, $(\mathcal{H} \otimes I_n)$ is symmetric with positive diagonal and non-positive off-diagonal elements. This means that, the matrix $(\mathcal{H} \otimes I_n)$ has positive determinant and positive eigenvalues. Hence, it is a non-singular M-matrix [54]–[56]. As a result, one can say that $(\mathcal{H} \otimes I_n) > 0$. Then, the first term in (46) is surely negative. To achieve $\dot{U}_3 < 0$, we should show that

$$U_{31} = \tau^T(\mathcal{H} \otimes I_n)\hat{s}_1 - \tau^T(\mathcal{H} \otimes \dot{s}^b)1 \leq 0. \tag{47}$$

Recalling the mixed-product property of Kronecker product, we have

$$(\mathcal{H} \otimes \dot{s}^b) = (\mathcal{H} \otimes I_n)(I_N \otimes \dot{s}^b). \tag{48}$$

Hence, (47) can be written as following

$$U_{31} = \tau^T(\mathcal{H} \otimes I_n)\hat{s}_1 - \tau^T(\mathcal{H} \otimes I_n)(I_N \otimes \dot{s}^b)1, \tag{49}$$

and then

$$U_{31} \leq \tau^T(\mathcal{H} \otimes I_n)\hat{s}_1 + ABS(\tau^T(\mathcal{H} \otimes I_n))(I_N \otimes M^b)1, \tag{50}$$

where for $v \in \mathbb{R}^{Nn\times 1}$, we define

$$ABS(v) = [|v(1)|, |v(2)|, \ldots, |v(Nn)|]^T. \tag{51}$$

Now, we should only show that

$$\tau^T(\mathcal{H}_s \otimes I_n)\hat{s}_1 + ABS(\tau^T(\mathcal{H} \otimes I_n))(I_N \otimes M^b)1 = 0. \tag{52}$$

Thus, we reach to

$$\tau^T(\mathcal{H} \otimes I_n)\hat{s}_1 = -ABS(\tau^T(\mathcal{H} \otimes I_n))(I_N \otimes M^b)1, \tag{53}$$

and then

$$\tau^T(\mathcal{H} \otimes I_n)\hat{s}_1 = -\tau^T(\mathcal{H} \otimes I_n)\mathcal{M}(sgn\{\tau^T(\mathcal{H} \otimes I_n)\})(I_N \otimes M^b)1, \tag{54}$$

where $\mathcal{M}(sgn\{\tau^T(\mathcal{H} \otimes I_n)\}) \in \mathcal{R}^{Nn\times Nn}$ is a diagonal matrix whose diagonal elements are the sign of each element in $\tau^T(\mathcal{H} \otimes I_n) \in \mathcal{R}^{1\times Nn}$. Finally, since

$$(\tau\tau^T(\mathcal{H} \otimes I_n))^{-1}\tau\tau^T(\mathcal{H} \otimes I_n) = I_N \otimes I_n, \tag{55}$$

we have

$$\hat{s}_1 = -\mathcal{M}(sgn\{\tau^T(\mathcal{H} \otimes I_n)\})(I_N \otimes M^b)1, \tag{56}$$

and then the rates for observed parameters are

$$\dot{\hat{s}} = -\lambda\tau - \mathcal{M}(sgn\{\tau^T(\mathcal{H} \otimes I_n)\})(I_N \otimes M^b)1. \tag{57}$$

By utilizing $\dot{\hat{s}}$ from (57), we can have $\dot{U}_3 < 0$, which in turn shows that the consensus error on observation (i.e. τ) is asymptotically stable and converges to zero referring to the Lyapunov stability theorem. Hence, the *Proposition 6* is achieved. Moreover, the observer for agent i can be represented as follows

$$\dot{\hat{s}}^i = -\lambda\tau^i - [\mathcal{M}(sgn\{\sum_{j=1}^{N}(\mathcal{H}(i,j)\tau^j\})\; M^b]. \tag{58}$$

Then, the equation in (41) is achieved by referring to (35) and (38). This completes the proof. ■

Remark 2: The values for M^b can be determined according to the actuator limits and also several previous experiments of the dynamic agent. For example, for p^b as the position of a mobile robot, M^b is the maximum absolute values for the mobile robot speed, which can be defined according to the actuators specifications and some data from previous field experiments.

Remark 3: The value of scalar gain λ should be large enough in order to reach fast finite-time convergence of the distributed estimation algorithm in (41).

V. PRESENTING THE ACL ALGORITHM

By combining the results provided in *Theorem 1* and *Theorem 2*, the ACL algorithm is presented in TABLE 1. As declared in this table, the input variables are all available locally at each agent, either by on-board measurements or by having the communication links with the neighboring agents.

TABLE 1. ACL algorithm.

Algorithm 1: ACL (at agent i in the network)

Initialization:
$\alpha \in (0,1]$, $\lambda = 100$, $M^b = 10 \times 1_n$

Main Loop:
for $\{t = d_t : d_t : t_f\}$ **do**
  Inputs: $\{\ p^b, V_r^{ij}(t), d_r^{ij}(t), V_r^{ib}(t), d_r^{ib}(t), \hat{p}^j(t-d_t), \tau^j(t-d_t), a(i,j), \beta_i, \mathcal{H}(i,j) \quad (j \neq i \in [1,N])\ \}$

  for $\{j = 1 : N\}$ **do**
   if $\{a(i,j) == 1\}$ **do**
1-1: $e^{ij}(t) = (d_r^{ij}(t))^2 - (\hat{P}_r^{ij}(t-d_t))^T \hat{P}_r^{ij}(t-d_t)$
1-2: $g^{ij}(t) = (V_r^{ij}(t))^T \hat{P}_r^{ij}(t-d_t)$
1-3: $\dot{\hat{P}}_r^{ij}(t) = [1 + \alpha\ sgn(e^{ij}(t)g^{ij}(t))]V_r^{ij}(t)$
1-4: $\delta_{ij}(t) = \delta_{ij}(t-d_t) + \{\dot{\hat{P}}_r^{ij}(t) \times d_t\}$
   end if
  end for
  if $\{\beta_i^b == 1\}$ **do**
2-1: $e^{ib}(t) = (d_r^{ib}(t))^2 - (\hat{P}_r^{ib}(t-d_t))^T \hat{P}_r^{ib}(t-d_t)$
2-2: $g^{ib}(t) = (V_r^{ib}(t))^T \hat{P}_r^{ib}(t-d_t)$
2-3: $\dot{\hat{P}}_r^{ib}(t) = [1 + \alpha\ sgn(e^{ib}(t)g^{ib}(t))]V_r^{ib}(t)$
2-4: $\delta_{ib}(t) = \delta_{ib}(t-d_t) + \{\dot{\hat{P}}_r^{ib}(t) \times d_t\}$
  end if
3: $\tau^i(t) = \beta_i^b\ [\hat{p}^i(t-d_t) - (p^b(t) + \delta ib(t))]$
  for $\{j = 1 : N\}$ **do**
4: $\tau^i(t) = \tau^i(t) + a(i,j)\ [\hat{p}^i(t-d_t) - (\hat{p}^j(t-d_t) + \delta ij(t))]$
  end for
5: $\dot{\hat{p}}^i(t) = -\lambda\tau^i(t)$
  for $\{j = 1 : N\}$ **do**
6: $\dot{\hat{p}}^i(t) = \dot{\hat{p}}^i(t) - \mathcal{M}(sgn\{\mathcal{H}(i,j)\tau^j(t-d_t)\}) \times M^b$
  end for
7: $\hat{p}^i(t) = \hat{p}^i(t-d_t) + \{\dot{\hat{p}}^i(t) \times d_t\}$
  Outputs: $\hat{p}^i(t)),\ \delta_{ij}(t),\ \tau^i(t)$
end for

The output variables in ACL algorithm include the estimated absolute position for each agent and the estimated relative position for each pair of connected agents in the network.

VI. SIMULATION STUDIES

In this section, three different case studies are presented to evaluate the ACL algorithm. In the first two cases, the performance of the ACL algorithm is compared with two recent localization solutions.

A. CASE STUDY-1: ONE MOBILE AGENT WITH ONE BEACON AGENT

Suppose that we have a mobile agent in 2D environment with a nonlinear double-integrator dynamics as follows

$$\begin{aligned} \dot{p} &= v \\ \dot{v} &= u - v\sin(0.2t) + 0.5p\cos(0.1t), \end{aligned} \tag{59}$$

where $p = [p_x; p_y] \in \mathbb{R}^2$, $v = [v_x; v_y] \in \mathbb{R}^2$ and $u = [u_x; u_y] \in \mathbb{R}^2$ are the position, velocity and the control input of the mobile agent, respectively. No on-board sensor to measure the absolute position on the mobile agent is assumed. Instead, the relative distance to a beacon agent can be measured using the UWB antennas located on both agents. Moreover, the velocity of the mobile agent can be computed using the on-board IMU module combined with a Kalman-filter [7]. In addition, the velocity and absolute position of the beacon agent are transmitted to the mobile agent via the existing communication link. The beacon agent has an on-board sensor to measure its own absolute position, i.e p_b. In this sense, the relative velocity and relative distance between the mobile agent and the beacon are available at the mobile agent. Thus, the estimation algorithm proposed in (5) can be used to estimate the relative position of the mobile agent to the beacon. After that, the estimated absolute position of the mobile agent $\hat{p}$ in a 2D environment can be computed online according to (32). In this study, the mobile agent uses a simple back-stepping controller [52] to satisfy the tracking objective for the dynamics model presented in (59), as follows

$$u = k_1^c e_p - k_2^c v, \tag{60}$$

where k_1^c and k_2^c are two positive constant scalars and $e_p = p_d - \hat{p}$ is the tracking error (p_d is the desired trajectory) based on the estimated positions. The beacon agent having the same dynamics as in (59), also uses the same controller as presented in (60); but its corresponding position tracking error is computed using the measured absolute values, i.e. p_b. Note that use of the controller in (60) can satisfy conditions on the boundedness of the relative velocity and the relative distance of the two agents requested in *Assumption 1*. Here, the initial value for estimated position of the mobile agent is set at [0; 0].

In order to evaluate the performance of the proposed adaptive relative position estimator in (5), the algorithm is compared with two other relative position estimation solutions.

TABLE 2. Case study-1: The values of tuning parameters for the controller and the relative positioning algorithms.

Desired Trajectory	k_1^c	k_2^c	β	α	Simulation time (sec)
$p_d^{(1)}$	10	10	0.001	1	200
$p_d^{(2)}$	10	100	0.001	0.01	3000
$p_d^{(3)}$	10	10	0.01	0.01	2000

The first solution has an intuitive formula as [40]

$$\dot{\hat{P}}_r = V_r. \tag{61}$$

The second solution is presented as follows [39]

$$\dot{\hat{P}}_r = [1 + \beta(2d_r\dot{d}_r - 2V_r^T\hat{P}_r)]V_r, \tag{62}$$

where $\beta \in \mathbb{R}^+$. The comparative study among the mentioned three algorithms (i.e. algorithms in (5), (61) and (62)) is presented in TABLE 3 and TABLE 4 for three different desired trajectories of the moving agent. These trajectories are a step input as $p_d^{(1)} = [5; -6]$, a square wave signal as $p_d^{(2)} = [Sq(5, 1000, 50); Sq(10, 1500, 50)]$; and also a sine wave as $p_d^{(3)} = [5\sin(0.1t); 3\sin(0.2t)]$. Here, $Sq(A_s, T_s, W_s)$ is a square wave signal with "A_s" amplitude, "T_s" duration and "W_s" pulse width percentage. Moreover, the study has been done for two cases with stationary and moving beacon agent. The desired position of the beacon agent is $p_d^b = [0; 0]$ for the first case with stationary beacon and $p_d^b = [2; -3]$ for the second simulation with moving beacon. Furthermore, the initial position of the moving agent is set at $p(0) = \hat{p}(0) = [0; 0]$ for all of the simulations and the parameters of the controller and the relative position estimation algorithms are tuned according to TABLE 2. Here, the cumulative values for positioning and tracking errors are considered for the comparison, as follows

$$C_1 = \int \xi^T \xi \, dt,$$
$$C_2 = \int e_r^T e_r \, dt, \tag{63}$$

where $\xi = p - \hat{p}$ is the absolute positioning error and $e_r = p_d - p$ is the tracking error based on the real absolute position of the mobile agent. As can be observed in TABLE 3 and TABLE 4, while a similar tracking controller is used for all of the algorithms, the proposed adaptive relative position estimating algorithm in (5) has lower values for C_1 and C_2 parameters and consequently outperforms the two other algorithms.

TABLE 3. Case study-1: Comparing the performance of three relative position estimation algorithms with a stationary beacon agent ($p_d^b = [0; 0]$).

Desired Trajectory	Eq. (61) C_1	C_2	Eq. (62) C_1	C_2	Eq. (5) C_1	C_2
$p_d^{(1)}$	73.38	115.7	42.5	88.42	0.1565	49.3
$p_d^{(2)}$	12.66	2894	11.14	2891	5.327	2883
$p_d^{(3)}$	2182	2584	86.22	476.2	28.05	462.1

TABLE 4. Case study-1: Comparing the performance of three relative position estimation algorithms with a moving beacon agent ($p_d^b = [2; -3]$).

Desired Trajectory	Eq. (61) C_1	C_2	Eq. (62) C_1	C_2	Eq. (5) C_1	C_2
$p_d^{(1)}$	20.23	66.9	18.69	65.49	0.1796	50.66
$p_d^{(2)}$	6.908	2887	6.872	2885	4.531	2887
$p_d^{(3)}$	1257	1647	463.2	831.2	56.62	455.6

B. CASE STUDY-2: A NETWORK OF NON-COOPERATIVELY CONTROLLED MOBILE AGENTS WITH ONE BEACON

Here, we have a network of four mobile agents with the dynamic system in (59) and one stationary beacon agent in a 2D environment. Here, each of the agents has a local controller as presented in (60) and it operates individually without any cooperative protocol in the network. In other words, these mobile agents move according to their individual desired trajectory oblivious of their neighboring agents. Similar to the first case study, here the mobile agents do not have any on-board sensor for measuring the absolute position. Instead, they communicate with the neighboring agents in the network so as to estimate their absolute position by using the ACL algorithm presented in TABLE 1. The relative distance and the relative velocity to the neighboring agents can be determined using the data provided via the communication graph. The communication graph of the network (FIGURE 1) satisfies the conditions requested in *Definition 2* and *Assumption 4*. The absolute position of the beacon agent is communicated to only one of the mobile agent via the provided communication link. In this regard, the corresponding adjacency and beacon pinning gain matrix for the communication graph of the network are defined as follows

$$\mathcal{A} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix}, \quad \mathcal{B} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}. \tag{64}$$

Here, the performance of the ACL algorithm is compared with a recently investigated linear-convex (LC) algorithm for localization in a network of mobile robots which is presented in [40]. That algorithm incorporates the triangulation method among the neighboring agents in the network and provides a trade-off between the triangulation and the intuitive relative position estimation proposed in (61). Based on [40], the position of the ith mobile agent can be estimated by the following adaptive law (to be updated at k step)

$$\hat{p}_{k+1}^i = \beta_0\hat{p}_k^i + (1 - \beta_0)[\sum_{j=1}^{3} a_k^{ij}\hat{p}_k^j] + v^i, \tag{65}$$

where $\beta_0 \in \mathbb{R}^+$ is a design parameter and it is non-zero if there is a triangulation set available around agent i. In addition, the values for $0 \leq a_k^{ij} \leq 1$ are the *barycentric* coordinates of agent i with respect to the neighboring agent j at step k and are computed as follows [40]

$$a_k^{ij} = \frac{A^j_{\Theta_i}(k)}{A_{\Theta_i}(k)}, \tag{66}$$

in which Θ_i is the triangulation set around agent i including its three neighboring agents. $A_{\Theta_i}(k)$ is the value for area of the triangle produced by the three neighboring agents at the kth step, while $A^j_{\Theta_i}(k)$ is the area of that triangle by removing the jth neighboring agent and replacing the agent i. The values for areas (or volumes in a 3D environment) are computed by exploiting the *Cayley-Menger* determinant [40]. The LC algorithm needs to have communication links to three neighboring agents in order to utilize the triangulation technique, otherwise the estimation would be driven with only the velocity measurements, i.e v^i. In this sense, for a network of 4 mobile agents and one stationary beacon agent, there will be a requirement of 7 undirected communication links among the agents in the network, including only one communication link between the beacon and agent-1. On the other hand, the ACL algorithm proposed in the current paper requires only 4 communication links (according to FIGURE 1). Having lower number of communication links can provide lower energy consumption at each agent and consequently for the whole network. Besides, as it is declared in TABLE 5, the tracking and positioning errors are much lower for the ACL algorithm in comparison with the LC algorithm suggested in [40]. Note that Here, the simulations are performed for tracking the square wave desired trajectory at each agent and the values for tuning parameters are set same as in the second row of TABLE 2. Also, we used $\beta_0 = 0.25$ as suggested in [40]. The simulation results for this comparative study are presented in FIGURE 2 to FIGURE 7. It is evident that, by combining the ACL algorithm and the stable controller in (60), absolute positions of the mobile agents converge to the desired values, asymptotically (FIGURE 4 and FIGURE 5). Moreover, the absolute positioning errors are much smaller using ACL algorithm in comparison with the ones provided by utilizing the LC algorithm (FIGURE 2 and FIGURE 3).

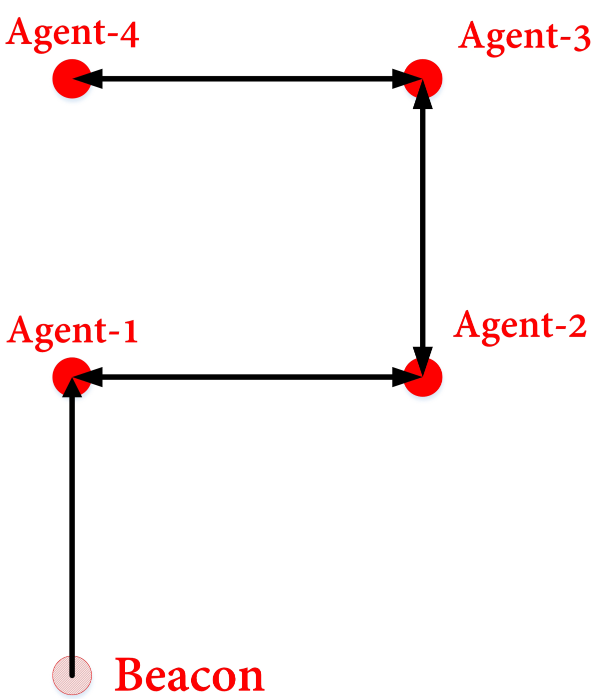


FIGURE 1. Case study-2: the communication graph among the agents in the network that the ACL algorithm is applied to.

TABLE 5. Case study-2: Comparing the performance of the ACL algorithm against the LC algorithm proposed in [40].

Agent	LC		ACL	
	C_1	C_2	C_1	C_2
Agent-1	4.447	156.1	0.0897	143.5
Agent-2	8.852	322.5	0.2683	299.4
Agent-3	10.7	241.4	0.4341	214.0
Agent-4	7.94	225.7	0.5311	207.7

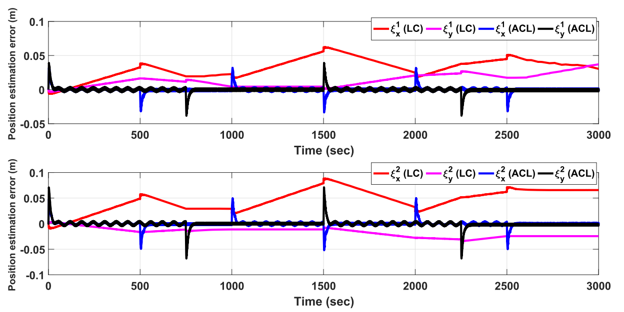


FIGURE 2. Case study-2: position estimation error for the mobile agents in the network (Agent-1 and Agent-2).

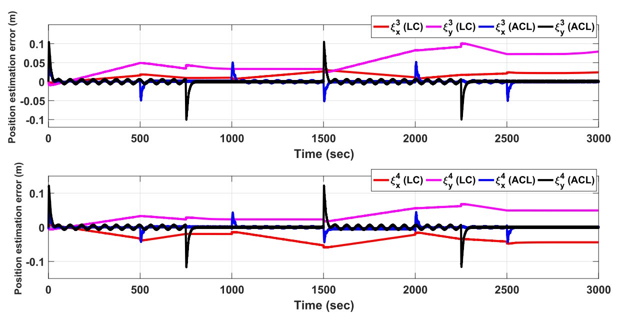


FIGURE 3. Case study-2: position estimation error for the mobile agents in the network (Agent-3 and Agent-4).

C. CASE STUDY-3: A NETWORK OF COOPERATIVELY CONTROLLED MOBILE AGENTS WITH ONE BEACON

In this section, performance of the ACL algorithm is evaluated in a network of cooperatively controlled mobile agents

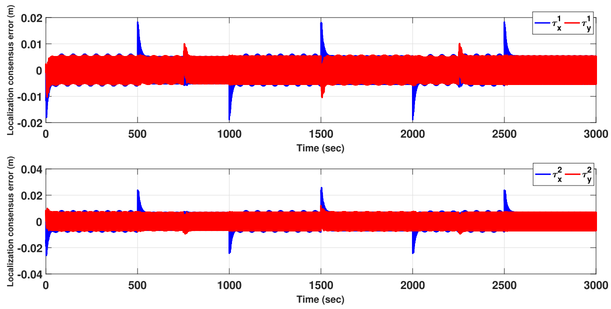


FIGURE 4. **Case study-2: consensus error of the cooperative observer for mobile agents in the network (Agent-1 and Agent-2).**

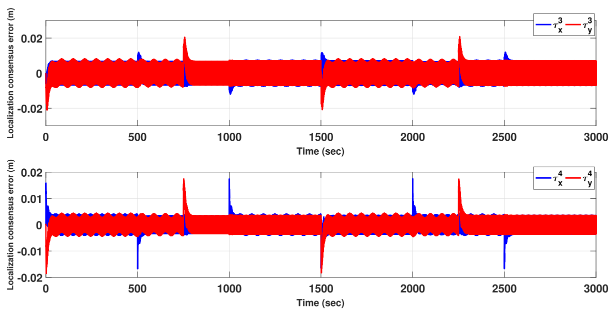


FIGURE 5. **Case study-2: consensus error of the cooperative observer for mobile agents in the network (Agent-3 and Agent-4).**

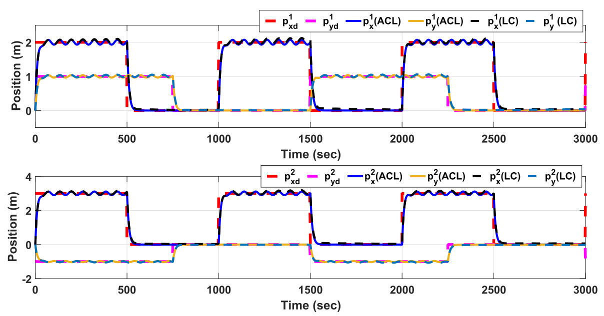


FIGURE 6. **Case study-2: absolute position of the mobile agents in the network (Agent-1 and Agent-2).**

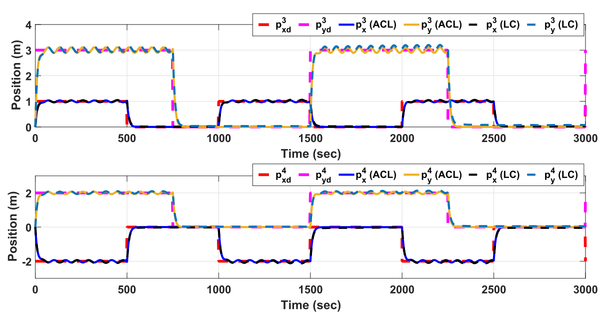


FIGURE 7. **Case study-2: absolute position of the mobile agents in the network (Agent-3 and Agent-4).**

in a 2D environment. In this sense, the agents with the same dynamic system presented in (59), use decentralized cooperative controllers for achieving the consensus objective in the network. Here, the properties of the network are same as the ones for the network in *Case study-2*. Thus, the adjacency and the beacon pinning gain matrices are similar to the ones presented in (64). Moreover, the beacon agent is supposed to have the similar dynamic system to (59) and it is controlled with an adaptive model-free controller as follows [45]

$$
\begin{aligned}
u^0 &= [u^0_{aux_1}; u^0_{aux_2}; u^0_x; u^0_y] = u^0_1 + u^0_2 \\
u^0_1 &= \frac{1}{2} R^0 (B^0)^T P^0 \sigma^0 \\
u^0_2 &= (B^0)^{-1}\{[\dot{p}^0_{des}; \dot{v}^0_{des}] - \hat{A}^0[p^0; v^0] - \hat{g}^0 - \zeta^0 \\
&\quad + (I_4 + 2(P^0)^{-1}Q^0 + \hat{A}^0)\sigma^0\} - \frac{3}{4} R^0 (B^0)^T P^0 \sigma^0.
\end{aligned} \tag{67}
$$

Here, the superscript 0 is used to designate a beacon as a reference node or a *leader* node by virtue of multi-agent theory. In (67), $I_4 \in \mathbb{R}^{4\times4}$ is the identity matrix, B^0, R^0 and Q^0 are constant positive definite matrices in $\mathbb{R}^{4\times4}$ and $P^0 = \{P^0\}^T \in \mathbb{R}^{4\times4}$ is a positive definite matrix defined by using [45]

$$
\dot{P}^0 = \{\hat{A}^0\}^T P^0 + P^0 \hat{A}^0 - P^0 B^0 R^0 \{B^0\}^T P^0 + 2Q^0. \tag{68}
$$

Moreover, the following adaptive laws are incorporated [45]

$$
\begin{aligned}
\dot{\hat{g}}^0 &= -\Gamma^0_1 P^0 \sigma^0 - \rho^0_1 \Gamma^0_1 \hat{g}^0 \\
\dot{v}_{\hat{A}^0} &= -\Gamma^0_2 P^0 \mathcal{M}(\sigma^0)([p^0; v^0] - \sigma^0) - \rho^0_2 \Gamma^0_2 v_{\hat{A}^0},
\end{aligned} \tag{69}
$$

for online estimation of the unknown nonlinear dynamics (the dynamics in (59) is assumed to be unknown from the controllers points of view), where Γ^0_1 and Γ^0_2 are two positive definite diagonal matrices in $\mathbb{R}^{4\times4}$ including the adaptive gains and ρ^0_1 and ρ^0_2 are two positive scalar leakage gains. We have $\sigma^0 = e^0_p + \zeta^0$, where e^0_p is the tracking error at the beacon agent and $\zeta^0 = \int e^0_p \; dt$. Noth that $v_{\hat{A}^0}$ is a vector whose elements are the diagonal entities in $\hat{A}^0$. The detailed convergence analysis of the controller and the adaptive laws proposed in (67) to (69) can be found in [44] and [45] for single-input single-output and multi-input multi-output nonlinear dynamic systems. Moreover, the applications of this control algorithm are presented in [46]–[48] for a chaotic resonator, a robotic manipulator and an underwater autonomous robot, respectively.

To provide consensus among the mobile agents, a model-free decentralized cooperative controller is used. The exploited controller at the ith agent in the network is defined as follows [50]

$$
\begin{aligned}
u^i &= \frac{1}{\mathcal{H}(i,i)}(B^T B)^{-1} B^T [-K_c P^i_c e^i_c - K_I \zeta^i_c - \mathcal{H}(i,i)\hat{g}^i_c \\
&\quad - \sum_{j=1 \neq i}^{N} \{\mathcal{H}(i,j)(\hat{T}^j_c - \hat{A}^j_c[\hat{p}^j; v^j] - 2\hat{g}^j_c)\} \\
&\quad + b_i(u^0 - \hat{A}^i_c[p^0; v^0])],
\end{aligned} \tag{70}
$$

where $B \in \mathbb{R}^{4\times4}$ is a positive definite matrix and

$$
\begin{aligned}
e_c^i &= \sum_{j=1}^{N} a_{ij}([\hat{p}^i; v^i] - [\hat{p}^j; v^j]) + b_i([\hat{p}^i; v^i] - [p^0; v^0]) \\
\epsilon_c^i &= \sum_{j=1}^{N} a_{ij}(\hat{T}_c^i - \hat{T}_c^j) + b_i(\hat{T}_c^i - u^0) \\
\dot{\hat{g}}_c^j &= \gamma_1 \sum_{k=1}^{N} (\mathcal{H}(k,j) P_c^k e_c^k) - \rho_1 \hat{g}_c^j \\
\dot{\hat{A}}_c^j &= \gamma_2 \tilde{w}^j r^{jT} - \rho_2 \gamma_2 \hat{A}_c^j \\
\dot{\hat{T}}_c^j &= -\mu \epsilon_c^j - [\mathcal{M}(sgn\{\sum_{k=1}^{N}(\mathcal{H}(j,k)\epsilon_c^k)\}) \times u_M^0] \\
\tilde{w}^i &= \frac{s}{s+1}[\hat{p}^i; v^i] - \frac{1}{s+1} B u^i - \frac{1}{s+1}\hat{g}_c^i - \hat{A}_c r^i \\
r^i &= \frac{1}{s+1}[\hat{p}^i; v^i] \qquad (71)
\end{aligned}
$$

where s is the symbol for Laplace variable in Laplace transform in the filter expression and P_c^i is solution of the following continuous algebraic Riccati equation

$$
(\hat{A}_c^i)^T P_c^i + P_c^i \hat{A}^i - P_c^i K_c^i P_c^i = -Q. \qquad (72)
$$

In above, K_c^i, K_I^i are positive definite matrices in $\mathbb{R}^{4\times4}$, $\mathcal{H}(i,j)$ is the element of $\mathcal{H}$ in ith row and jth column and $\zeta_c^i = \int e_c^i dt \in \mathbb{R}^{4\times1}$. Moreover, μ, γ_1 and γ_2 are positive constant scalars defining the adaptation rates, while ρ_1 and ρ_2 are another positive scalars acting as the leakage gains in the cooperative adaptive laws. The detailed proof of the above controller is presented in [49] and [50].

The desired path that the beacon should track and all of the other agents in the network must follow is $p_{des}^0 = [3\sin\ 0.1\ t; 6\sin\ 0.1\ t]$. The values for constant parameters in the ACL algorithm, the model-free controller at the beacon agent and the model-free decentralized cooperative controller at the mobile agents are presented in TABLE 6. The simulation results for the current case study are depicted in FIGURE 8 to FIGURE 15. Appropriate performance of the algorithms can be observed and the consensus over the desired

TABLE 6. Case study-3: the constant parameters for the ACL algorithm, the model-free controller at the beacon agent and the model-free decentralized controller at the other agents in the network.

Parameter	Value	Parameter	Value
α	0.01		
λ	10	M^b	[0.1;0.1]
$B^0 = R^0$	I_4	Q^0	$0.1 \times I_4$
Γ_1^0	$diag([0.01; 0.01; 10; 10])$	ρ_1^0	0.1
Γ_2^0	$diag([0.001; 0.001; 1; 1])$	ρ_2^0	0.1
B	I_4	Q	I_4
K_c	$1000 \times I_4$	K_I	$0.1 \times I_4$
γ_1	1	ρ_1	1000
γ_2	1	ρ_2	1
μ	100	u_M^0	[1;1;1;1]

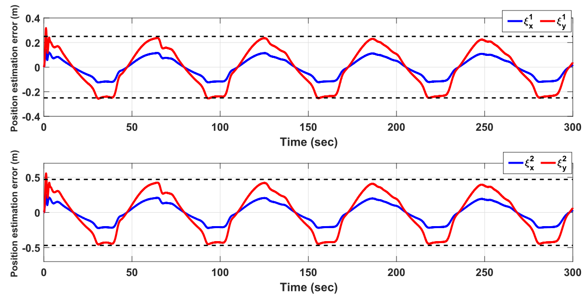


FIGURE 8. Case study-3: position estimation error for the mobile agents in the network (Agent-1: bounded in [−0.25,0.25]; and Agent-2:: bounded in [−0.47,0.47]).

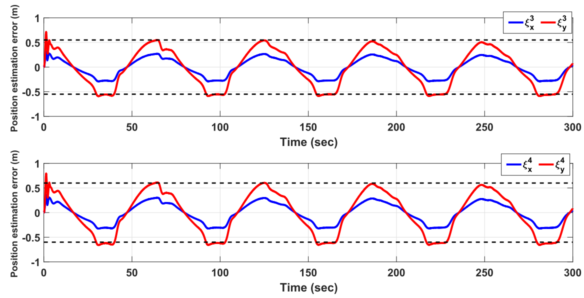


FIGURE 9. Case study-3: position estimation error for the mobile agents in the network (Agent-3: bounded in [−0.55,0.55]; and Agent-4 bounded in [−0.60,0.60]).

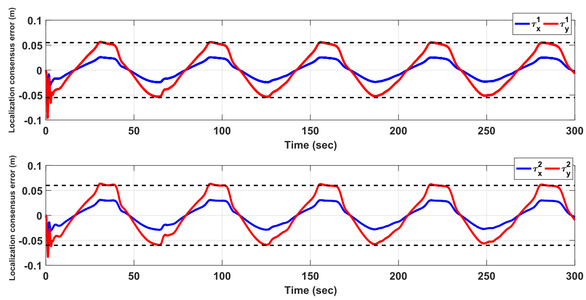


FIGURE 10. Case study-3: localization consensus error for the mobile agents in the network (Agent-1: bounded in [−0.055,0.055]; and Agent-2: bounded in [−0.060,0.060]).

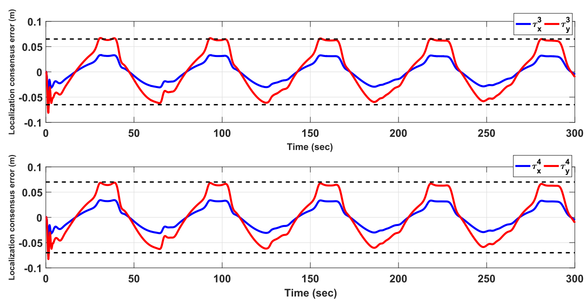


FIGURE 11. Case study-3: localization consensus error for the mobile agents in the network (Agent-3: bounded in [−0.065,0.065]; and Agent-4: bounded in [−0.070,0.070]).

position is achieved in the network by combination of the cooperative localization algorithm and the model-free cooperative controller. In addition, it is shown that the position

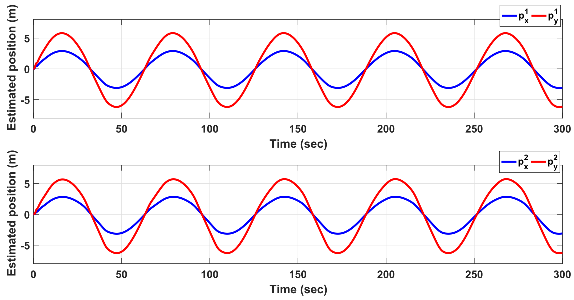


FIGURE 12. Case study-3: estimated absolute position of the agents in the network (Agent-1 and Agent-2).

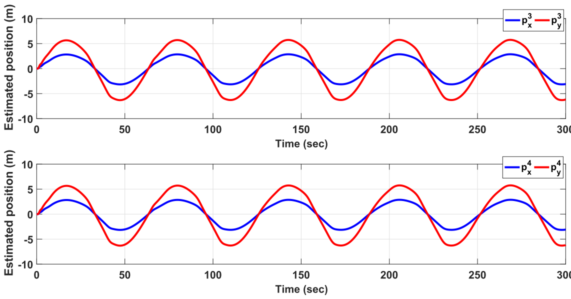


FIGURE 13. Case study-3: estimated absolute position of the agents in the network (Agent-3 and Agent-4).

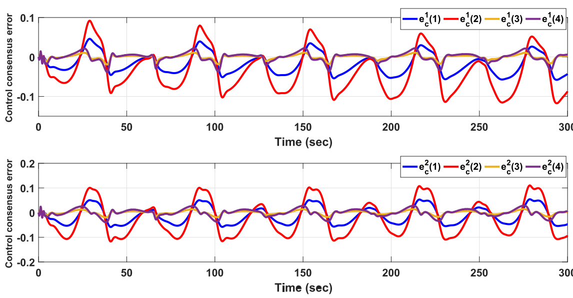


FIGURE 14. Case study-3: control consensus errors at the agents in the network (Agent-1 and Agent-2).

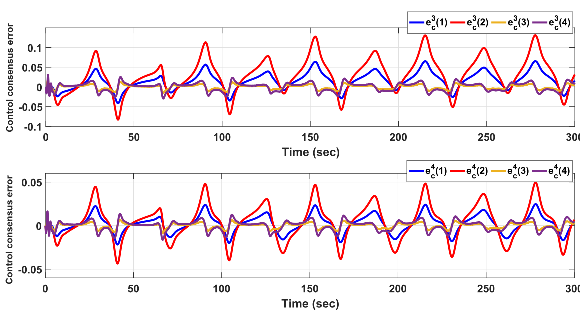


FIGURE 15. Case study-3: control consensus errors at the agents in the network (Agent-3 and Agent-4).

estimation errors (i.e. e_p^is) in FIGURE 8 and FIGURE 9, the consensus localization errors (i.e. τ^is) in FIGURE 10 and FIGURE 11 as well as the control consensus errors (i.e. e_c^is) in FIGURE 14 and FIGURE 15 are all bounded in the small sets around zero.

VII. CONCLUSION

In this paper, an adaptive relative position estimator is incorporated in a cooperative observer to provide a solution for the cooperative localization problem in a network including one or more mobile agents. The solution does not depend on the dynamics of the agents and the local control signals. The algorithm needs each pair of the mobile agents to have non-zero relative velocities and non-zero relative distances. In contrast to the recent established work, the proposed algorithm does not require the measurement of the communicated signals' angle of arrival. In addition, the absolute positions of the mobile agents can be estimated by accessing to only one beacon agent in the network. There is only a requirement for an undirected communication path between each two agents in the communication graph of the network to confirm the stability and convergence of the solution. This is the least possible requirement for a spanning-tree communication graph, which leads to more convenience for practical implementations. According to the provided simulation studies, the adaptive relative position estimator works well for each pair of the mobile agents and it outperforms two other state-of-the-art relative position estimation algorithms. Furthermore, it is shown that the proposed ACL algorithm has better performance in comparison with the LC algorithm which exploits the triangulation technique among the three neighboring agents in a network. Besides, the ACL algorithm requires less communication links among the agents, while the LC algorithm needs the maximum number of links to the neighboring agents. In addition, the appropriate performance for the ACL algorithm has been observed in a team of mobile agents, which are cooperatively controlled using a decentralized adaptive model-free cooperative controller. The proven results promise a profound advantage especially in the context of a network of autonomous mobile robots application in a remote place such as search and rescue (SAR) mission when GPS signal is degraded and often obscured.

ACKNOWLEDGMENT

The PhD studies of the first author is under a TWAS-USM Postgraduate Fellowship.

REFERENCES

[1] F. van Diggelen and P. Enge, "The worlds first gps mooc and worldwide laboratory using smartphones," in *Proc. 28th Int. Tech. Meeting Satell. Division Inst. Navigat. (ION GNSS+)*, Tampa, Florida, Sep. 2015, pp. 361–369.

[2] R. W. Hedgecook, "Precise real-time localization using single-frequency GPS," Ph.D. dissertation, Dept. Elect. Eng., Fac. Graduate School Vanderbilt Univ., Nashville, TN, USA, 2014.

[3] P. A. Zandbergen, "Accuracy of iPhone locations: A comparison of assisted GPS, WiFi and cellular positioning," *Trans. GIS*, vol. 13, no. s1, pp. 5–26, 2009.

[4] M. Kok, J. D. Hol, and T. B. Schön. (2017). "Using inertial sensors for position and orientation estimation." [Online]. Available: https://arxiv.org/abs/1704.06053

[5] R. Tron, J. Thomas, G. Loianno, K. Daniilidis, and V. Kumar, "A distributed optimization framework for localization and formation control: Applications to vision-based measurements," *IEEE Control Syst. Mag.*, vol. 36, no. 4, pp. 22–44, Aug. 2016.

[6] J. S. Russel, M. Ye, B. D. O. Anderson, H. Hmam, and P. Sarunic, "Cooperative localisation of a GPS-denied UAV in 3-Dimensional space using direction of arrival measurements," *IFAC-PapersOnline*, vol. 50, no. 1, pp. 8019–8024, 2017.

[7] S. S. Kia, S. Rounds, and S. Martinez, "Cooperative localization for mobile agents: A recursive decentralized algorithm based on Kalman-filter decoupling," *IEEE Control Syst. Mag.*, vol. 36, no. 2, pp. 86–101, Apr. 2016.

[8] N. Bouzera, M. Oussalah, N. Mezhoud, and A. Khireddine, "Fuzzy extended Kalman filter for dynamic mobile localization in urban area using wireless network," *Appl. Soft Comput.*, vol. 57, pp. 452–467, Aug. 2017.

[9] J. Aspnes *et al.*, "A theory of network localization," *IEEE Trans. Mobile Comput.*, vol. 5, no. 12, pp. 1663–1678, Dec. 2006.

[10] G. Mao, B. Fidan, and B. D. O. Anderson, "Wireless sensor network localization techniques," *Comput. Netw.*, vol. 51, no. 10, pp. 2529–2553, 2007.

[11] A. Boukerche, H. A. B. F. Oliveira, E. F. Nakamura, and A. A. F. Loureiro, "Localization systems for wireless sensor networks," *IEEE Wireless Commun.*, vol. 14, no. 6, pp. 6–12, Dec. 2007.

[12] L. Chelouah, F. Semchedine, and L. Bouallouche-Medjkoune, "Localization protocols for mobile wireless sensor networks: A survey," *Comput. Elect. Eng.*, vol. 71, pp. 733–751, Oct. 2018.

[13] A. Stanoev, S. Filiposka, V. In, and L. Kocarev, "Cooperative method for wireless sensor network localization," *Ad Hoc Netw.*, vol. 40, pp. 61–72, Apr. 2016.

[14] N. B. Priyantha, H. Balakrishnan, E. D. Demaine, and S. J. Teller, "Anchor-free distributed localization in sensor networks," in *Proc. 1st Int. Conf. Embedded Netw. Sensor Syst. (SCM)*, 2003, pp. 340–341.

[15] H. Chenji and R. Stoleru, "Toward accurate mobile sensor network localization in noisy environments," *IEEE Trans. Mobile Comput.*, vol. 12, no. 6, pp. 1094–1106, Jun. 2013.

[16] S. Halder and A. Ghosal, "A survey on mobility-assisted localization techniques in wireless sensor networks," *J. Netw. Comput. Appl.*, vol. 60, pp. 82–94, Jan. 2016.

[17] Z. Lin, M. Fu, and Y. Diao, "Distributed self localization for relative position sensing networks in 2D space," *IEEE Trans. Signal Process.*, vol. 63, no. 14, pp. 3751–3761, Jul. 2015.

[18] S. Zhao and D. Zelazo, "Localizability and distributed protocols for bearing-based network localization in arbitrary dimensions," *Automatica*, vol. 69, pp. 334–341, Jul. 2016.

[19] G. Chai, C. Lin, Z. Lin, and M. Fu, "Single landmark based collaborative multi-agent localization with time-varying range measurements and information sharing," *Syst. Control Lett.*, vol. 87, pp. 56–63, Jan. 2016.

[20] T. R. Wanasinghe, G. K. I. Mann, and R. G. Gosine, "Distributed leader-assistive localization method for a heterogeneous multirobotic system," *IEEE Trans. Autom. Sci. Eng.*, vol. 12, no. 3, pp. 795–809, Jul. 2015.

[21] R. Kurazume, S. Nagata, and S. Hirose, "Cooperative positioning with multiple robots," in *Proc. IEEE Int. Conf. Robot. Automat.*, San Diego, CA, USA, May 1994, pp. 1250–1257.

[22] I. Rekleitis, G. Dudek, and E. Milios, "Probabilistic cooperative localization and mapping in practice," in *Proc. IEEE Int. Conf. Robot. Automat.*, Taipei, Taiwan, Sep. 2003, pp. 1907–1912.

[23] I. Rekleitis, G. Dudek, and E. Milios, "Experiments in free-space triangulation using cooperative localization," in *Proc. IEEE/RSJ Int. Conf. Intell. Robot. Syst. (IROS)*, Las Vegas, NV, USA, Oct. 2003, pp. 1777–1782.

[24] N. Trawny and T. Barfoot, "Optimized motion strategies for cooperative localization of mobile robots," in *Proc. IEEE Int. Conf. Robot. Automat. (ICRA)*, New Orleans, LA, USA, Apr./May 2004, pp. 1027–1032.

[25] S. I. Roumeliotis and G. A. Bekey, "Collective localization: A distributed Kalman filter approach to localization of groups of mobile robots," in *Proc. IEEE Int. Conf. Robot. Automat. (ICRA)*, San Francisco CA, USA, Apr. 2000, pp. 2958–2965.

[26] S. I. Roumeliotis and G. A. Bekey, "Distributed multirobot localization," *IEEE Trans. Robot. Autom.*, vol. 18, no. 5, pp. 781–795, Oct. 2002.

[27] E. D. Nerurkar, S. I. Roumeliotis, and A. Martinelli, "Distributed maximum a posteriori estimation for multi-robot cooperative localization," in *Proc. IEEE Int. Conf. Robot. Automat.*, Kobe, Japan, May 2009, pp. 1402–1409.

[28] D. Fox, W. Burgard, H. Kruppa, and S. Thrun, "A probabilistic approach to collaborative multi-robot localization," *Auto. Robots*, vol. 8, no. 3, pp. 325–344, 2000.

[29] N. E. Özkucur, B. Kurt, and H. L. Akin, "A collaborative multi-robot localization method without robot identification," in *RoboCup 2008: Robot Soccer World Cup XII* (Lecture Notes in Computer Science), vol. 5399. Berlin, Germany: Springer, 2009, pp. 189–199.

[30] R. Huang and G. V. Záruba, "Monte Carlo localization of wireless sensor networks with a single mobile beacon," *Wireless Netw.*, vol. 15, no. 8, pp. 978–990, 2009.

[31] A. Prorok and A. Martinoli, "A reciprocal sampling algorithm for lightweight distributed multi-robot localization," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, San Fransisco, CA, USA, Sep. 2011, pp. 3241–3247.

[32] A. Prorok, A. Bahr, and A. Martinoli, "Low-cost collaborative localization for large-scale multi-robot systems," in *Proc. IEEE Int. Conf. Robot. Automat.*, Saint Paul, MN, USA, May 2012, pp. 4236–4241.

[33] K. K. Oh and H. S. Ahn, "Formation control of mobile agents based on distributed position estimation," *IEEE Trans. Autom. Control*, vol. 58, no. 3, pp. 737–742, Mar. 2013.

[34] K. K. Oh, M. C. Park, and H. S. Ahn, "A survey of multi-agent formation control," *Automatica*, vol. 53, pp. 424–440, Mar. 2015.

[35] F. Morbidi and G. L. Mariottini, "Active target tracking and cooperative localization for teams of aerial vehicles," *IEEE Trans. Control Syst. Technol.*, vol. 21, no. 5, pp. 1694–1707, Sep. 2013.

[36] J. Lu, C. Li, and W. Su, "Extended Kalman filter based localization for a mobile robot team," in *Proc. Chin. Control Decis. Conf.*, Yinchuan, China, May 2016, pp. 939–944.

[37] S. Wang, D. Gu, L. Chen, and H. Hu, "Single beacon-based localization with constraints and unknown initial poses," *IEEE Trans. Ind. Electron.*, vol. 63, no. 4, pp. 2229–2241, Apr. 2016.

[38] F. Meyer, O. Hlinka, H. Wymeersch, E. Riegler, and F. Hlawatsch, "Distributed localization and tracking of mobile networks including noncooperative objects," *IEEE Trans. Signal Inf. Process. Net.*, vol. 2, no. 1, pp. 57–71, Mar. 2016.

[39] Z. Han, K. Guo, L. Xie, and Z. Lin, "Integrated relative localization and leader–follower formation control," *IEEE Trans. Autom. Control*, vol. 64, no. 1, pp. 20–34, Jan. 2019. doi: 10.1109/TAC.2018.2800790.

[40] S. Safavi and U. A. Khan, "An opportunistic linear–convex algorithm for localization in mobile robot networks," *IEEE Trans. Robot.*, vol. 33, no. 4, pp. 875–888, Aug. 2017.

[41] S. Safavi, U. A. Khan, S. Kar, and J. M. F. Moura, "Distributed localization: A linear theory," *Proc. IEEE*, vol. 106, no. 7, pp. 1204–1223, Jul. 2018.

[42] S. O. H. Madgwick, A. J. L. Harrison, and R. Vaidyanathan, "Estimation of IMU and MARG orientation using a gradient descent algorithm," in *Proc. IEEE Int. Conf. Rehabil. Robot.*, Zürich, Switzerland, Jun./Jul. 2011, pp. 1–7.

[43] Y. Diao, Z. Lin, and M. Fu, "A barycentric coordinate based distributed localization algorithm for sensor networks," *IEEE Trans. Signal Process.*, vol. 62, no. 18, pp. 4760–4771, Sep. 2014.

[44] A. Safaei and M. N. Mahyuddin, "Adaptive model-free control based on an ultra-local model with model-free parameter estimations for a generic SISO system," *IEEE Access*, vol. 6, pp. 4266–4275, 2018.

[45] A. Safaei and M. N. Mahyuddin, "Optimal model-free control for a generic MIMO nonlinear system with application to autonomous mobile robots," *Int. J. Adapt. Control Signal Process.*, vol. 32, no. 6, pp. 792–815, 2018.

[46] A. Safaei and M. N. Mahyuddin, "An optimal adaptive model-free control with a Kalman-filter-based observer for a generic nonlinear MIMO system," in *Proc. IEEE 2nd Int. Conf. Autom. Control Intell. Syst. (I2CACIS)*, Kota Kinabalu, Malaysia, Oct. 2017, pp. 56–61.

[47] A. Safaei, Y. C. Koo, and M. N. Mahyuddin, "Adaptive model-free control for robotic manipulators," in *Proc. IEEE Int. Symp. Robot. Intell. Sensors (IRIS)*, Ottawa, ON, Canada, Oct. 2017, pp. 7–12.

[48] A. Safaei and M. N. Mahyuddin, "Application of the optimal adaptive model-free control algorithm on an autonomous underwater vehicle," in *Proc. IEEE 3rd Int. Conf. Adv. Robot. Mechatron. (ICARM)*, Singapore, Jul. 2018, pp. 346–350.

[49] A. Safaei and M. N. Mahyuddin, "Adaptive model-free consensus control for a network of nonlinear agents under the presence of measurement noise," in *Proc. 11th Asian Control Conf. (ASCC)*, Gold Coast, QLD, Australia, Dec. 2017, pp. 1701–1706.

[50] A. Safaei and M. N. Mahyuddin, "Distributed adaptive model-free cooperative control for a network of generic unknown nonlinear systems," *Int. J. Adv. Robotic Syst.*, vol. 15, no. 5, pp. 1–23, 2018.

[51] A. Safaei and M. N. Mahyuddin, "Distributed observer for a team of autonomous underwater vehicles utilizing a beacon unit on the surface," in *Proc. IEEE 7th Int. Conf. Underwater Syst. Technol., Theory Appl. (USYS)*, Kuala Lumpur, Malaysia, Dec. 2017, pp. 1–5.

[52] H. K. Khalil, *Nonlinear Systems*, 3rd ed. Upper Saddle River, NJ, USA: Prentice-Hall, 2002.

[53] N. Fischer, R. Kamalapurkar, and W. E. Dixon, "LaSalle-Yoshizawa corollaries for nonsmooth systems," *IEEE Trans. Autom. Control*, vol. 58, no. 9, pp. 2333–2338, Sep. 2013.

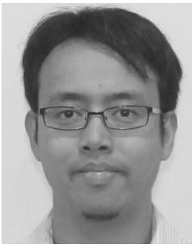
[54] Z. Li and Z. Duan. *Cooperative Control of Multi-Agent Systems*. Boca Raton, FL, USA: CRC Press, 2015.

[55] F. L. Lewis, H. Zhang, K. Hengster-Movric, and A. Das, *Cooperative Control of Multi-Agent Systems*. London, U.K.: Springer-Verlag, 2014.

[56] S. Boyd, L. El Ghaoui, E. Feron, and V. Balakrishnan, *Linear Matrix Inequalities in System and Control Theory*. Philadelphia, PA, USA: SIAM, 1994.

[57] A. Levant, "Higher-order sliding modes, differentiation and output-feedback control," *Int. J. Control*, vol. 76, nos. 9–10, pp. 924–941, 2003.

ALI SAFAEI received the B.Sc. degree in mechanical engineering from the Isfahan University of Technology, in 2009, and the M.Sc. degree in mechanical engineering from the University of Tehran, Iran, in 2012. He is currently pursuing the Ph.D. degree with the School of Electrical and Electronic Engineering, Universiti Sains Malaysia, Pulau Pinang, Malaysia. He received a prestigious TWAS-USM postgraduate fellowship for his Ph.D. studies. His research interests include adaptive model-free control, cooperative formation control, and cooperative localization algorithms for autonomous mobile robots.



MUHAMMAD NASIRUDDIN MAHYUDDIN (SM'18) the B.Eng. degree (Hons.) in mechatronic engineering from the International Islamic University of Malaysia, in 2004, the M.Eng. degree (Hons.) in mechatronic and automatic control from Universiti Teknologi Malaysia, in 2006, and the Ph.D. degree in mechanical engineering, specializing in control and robotics, from the University of Bristol, in 2014. Soon after graduation, he started his industrial career, in 2004, as an Application Engineer at Agilent Technologies, working with motion control product. He was appointed as a Senior Associate Teacher by the University of Bristol via contract, giving lecture in nonlinear control with application to robotics, from 2011 to 2012, and involved in a research project funded by Jaguar Land Rover. He was invited as a Visiting Professor with the MIS Lab, Universite de Picardie Jules Verne, France, in 2018. He was also attached to continental automotive components during his sabbatical, in 2019, working on closed-loop vehicle instrument cluster test. He is currently a Senior Lecturer with the School of Electrical and Electronics Engineering, Universiti Sains Malaysia, and also an Honorary Visiting Fellow with the Faculty of Engineering, University of Bristol. His current research interests include distributed adaptive control, cooperative control, nonlinear control, and parameter estimation involving mechatronics systems and robotics.