

Article

Multi-Vehicle Navigation Using Cooperative Localization

Juan Carlos Oliveros [†] and Hashem Ashrafiuon ^{*,†}

Center for Nonlinear Dynamics and Control, Department of Mechanical Engineering, Villanova University, 800 Lancaster Ave., Villanova, PA 19085, USA; jolivero@villanova.edu

* Correspondence: hashem.ashrafiuon@villanova.edu

† These authors contributed equally to this work.

Abstract: This paper assesses the effectiveness of cooperative localization for improving the performance of closed-loop control systems for networks for autonomous multi-vehicle navigation. Nonlinear dynamic models of two- and three-dimensional vehicles are presented along with their linearized forms. A nonlinear control algorithm is then presented based on the dynamic model. Relative position measurement equations and their linearized forms are introduced. The state and measurement equations are then employed for the propagation and update steps of an EKF-based cooperative localization algorithm. Initially, a series of experiments with networks of quadcopters and mobile robots are presented to validate the performance of cooperative localization for state estimation with the continuous or intermittent presence of absolute measurements or their complete absence. Finally, the performance of the control algorithm is evaluated with and without cooperative localization to demonstrate its effectiveness for improving performance.

Keywords: cooperative localization; multi-vehicle navigation; extended Kalman filter; autonomous navigation



Citation: Oliveros, J.C.; Ashrafiuon, H. Multi-Vehicle Navigation Using Cooperative Localization. *Electronics* **2023**, *12*, 4945. <https://doi.org/10.3390/electronics12244945>

Academic Editor: Mahmut Reyhanoglu

Received: 15 November 2023

Revised: 1 December 2023

Accepted: 7 December 2023

Published: 8 December 2023



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

1. Introduction

1.1. Motivation

Autonomous mapping, localization, and navigation refer to the abilities of a vehicle to create a map of the surrounding environment, localize itself on that map, and conduct a navigation plan. To achieve localization, an autonomous vehicle must be able to determine its position in the environment. Localization and navigation in multi-agent systems further complicate the problem since each vehicle may have its own set of sensors and algorithms that possibly create conflicting data and decisions about how to navigate through the environment. One common challenge in multi-vehicle systems is coordination. Each vehicle needs to be able to communicate with the others to avoid collisions and coordinate their movements. This requires sophisticated communication protocols and algorithms to ensure that each vehicle has the most up-to-date information about the other vehicles' locations and movements. Another challenge is scalability. As the number of vehicles in the system increases, the complexity of coordination and navigation increases as well. This can lead to issues such as congestion, bottlenecks, and communication delays, which can impact the performance and safety of the system. Addressing these challenges requires sophisticated algorithms and communication protocols as well as careful planning and testing to ensure the safe and efficient operation of the system.

Cooperative closed-loop control of multi-vehicle swarms has received significant attention due to military and civilian industry applications. Effective trajectory planning and cooperative closed-loop control of multi-vehicle systems, however, requires accurate localization of the agents to perform collaborative missions. In these applications, most if not all vehicles may not have access to accurate estimates of their own positions. It is therefore essential for vehicles to cooperate in localizing themselves. Moreover, if even a single vehicle can obtain an accurate estimate of its position and act as a reference base,

other members of the group should be able to evaluate their positions through relative position measurements and exchange of information.

Cooperative localization (CL) [1] enables vehicles to collaborate and share sensor measurements. By fusing these measurements together and leveraging relative positioning techniques, the vehicles can improve their localization accuracy, enabling safer and more reliable autonomous navigation, particularly in unstructured environments where methods such as feature-based SLAM [2] may not be effective. Moreover, CL can aid in reducing the sample rate requirement for SLAM methods in structured environments since these methods require extensive computation. There are also innovative control strategies [3,4] that showcase potential to enhance cooperative localization in multi-agent vehicle networks.

1.2. Related Works

Cooperative localization can utilize several techniques, of which EKF-based methods are the most common. In one of the earliest applications of CL, Roumeliotis [5] used an EKF algorithm that distributed information at the propagation step while keeping track of the cross-covariances created by relative measurements of each vehicle's pose. However, this method requires a completely synchronous and inter-connected communication network. Kia [1] proposed an Interim Master Decoupled centralized CL algorithm that eliminates the need for complete and constant communication between vehicles. It is assumed that any vehicle uses onboard sensors and its equations of motion to estimate its own pose through propagation and that any vehicle can measure the pose of nearby vehicles relative to its own reference frame. Vehicles are also assumed to be equipped with the capability to process this information and communicate with one another. CL allows the group of vehicles to cooperatively approximate an individual vehicle's pose by sharing the relative and absolute pose measurements that each vehicle has obtained [6].

In order to avoid the computational bottlenecks related to centralized algorithms, distributed or decentralized methods have been investigated [7–10]. These methods [11–14] have shown to be effective and efficient at the cost of accuracy. Centralized–decentralized techniques have shown great promise in improving localization with no knowledge of the environment [1,15]. A distributed CL utilizing EKF with low communication path requirements can also provide centralized-equivalent performance [16]. It is also possible to model the CL problem as a graph model to further utilize additional algorithms [17].

Cooperative localization methods have focused primarily on maneuvering in 2D space [18–22] and particularly using wheeled mobile robots [23–25] and vehicles [26,27] due to the complexities involved in the singularity-free representation of 3D orientations. It is, however, critically important to apply CL to 3D vehicle models to improve the localization of networks with homogeneous and heterogeneous air, marine, and ground vehicles. We have recently extended CL to such networks using both the commonly used kinematic model [28] and the newly developed dynamic model [29].

Exploring the synergy between cooperative localization, specifically employing simple and efficient Kalman filters [30], and advanced control strategies holds immense potential for achieving enhanced localization accuracy in multi-vehicle systems. Various strategies such as decentralized control frameworks [31], trajectory tracking [32], consensus algorithms [33], and swarm-based guidance systems [3] emphasize the pivotal role of cooperative localization in augmenting autonomous navigation capabilities in multi-vehicle networks.

1.3. Main Contributions and Novelties

This paper's main contribution is to better understand the impact of cooperative localization on performance in closed-loop control systems for autonomous vehicle networks. The main two contributions and novelties of this publication are outlined as follows:

1. This work represents the first application of cooperative localization in nonlinear closed-loop control of networks of 2D and 3D vehicles.
2. This work represents the first experimental validation of cooperative localization in closed-loop control of 2D and 3D vehicle networks.

- This work provides new guidance for optimal utilization of cooperative localization in state estimation for closed-loop control.

The paper is organized as follows. The vehicle model is presented in Section 2. The closed-loop control algorithm is presented in Section 3 followed by the multi-vehicle system model in Section 4 and the EKF-based CL algorithm in Section 5. All results, including for the validation and closed-loop control experiments, are presented in Section 6, followed by the concluding remarks in Section 7.

2. Single-Vehicle Model

2.1. 3D Vehicle Pose

The body-fixed reference frame $X_v Y_v Z_v$ is assigned to each vehicle, and its pose is represented by its origin's position and its orientation with respect to the inertial reference frame $X_0 Y_0 Z_0$, as shown for an air vehicle in Figure 1. The position of the vehicle is denoted by the vector $r_0^v = [x_0^v y_0^v z_0^v]^T$, whereas its orientation is determined by the orthogonal rotation matrix R_0^v , which maps the coordinates of a vector from $X_v Y_v Z_v$ to $X_0 Y_0 Z_0$. Quaternions are used to compute R_0^v to avoid singularities associated with Euler angles.

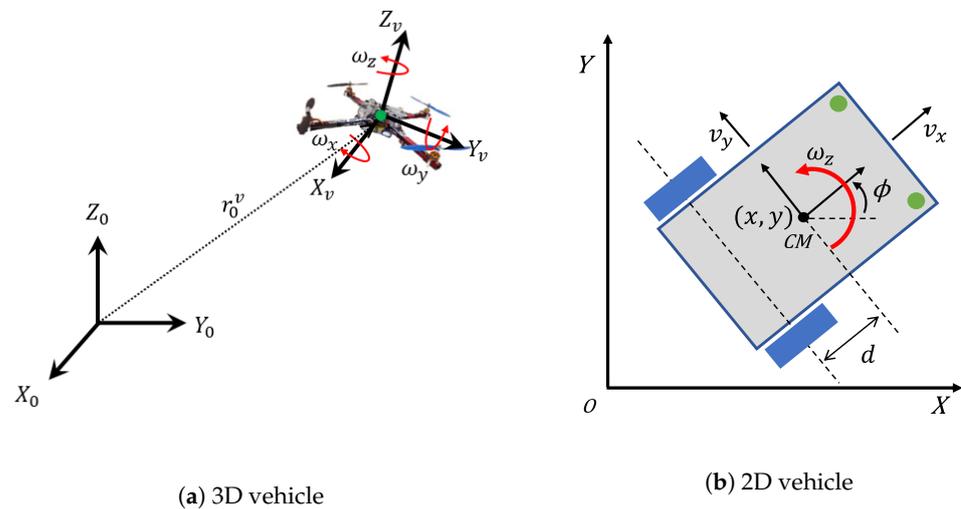


Figure 1. Kinematic models for 3D air vehicles (a) and 2D ground vehicles (b).

The quaternion vector \bar{q}^v consists of 4 components:

$$\bar{q}^v = \begin{bmatrix} q_0^v \\ q_1^v \\ q_2^v \\ q_3^v \end{bmatrix}, \quad q^v = \begin{bmatrix} q_1^v \\ q_2^v \\ q_3^v \end{bmatrix}. \tag{1}$$

The quaternions form a unit vector, i.e., $\bar{q}^{vT} \bar{q}^v - 1 = 0$, since there are only 3 rotational degrees of freedom. Thus, the vehicle's pose is represented by 7 components of r_0^v and \bar{q}^v .

Quaternions are not physical parameters to be measured and must be derived from R_0^v [28], which can be constructed using measured variables such as Euler angles. The orthogonal rotation matrix R_0^v in terms of quaternions [34] is defined as

$$R_0^v(\bar{q}^v) = [2(q_0^v)^2 - 1]I_3 + 2[q^v q^{vT} + q_0^v S(q^v)] \tag{2}$$

where I_3 is the 3×3 identity matrix and $S(q^v)$ defines a skew-symmetric matrix of a vector such that

$$S(q^v) = \begin{bmatrix} 0 & -q_3^v & q_2^v \\ q_3^v & 0 & -q_1^v \\ -q_2^v & q_1^v & 0 \end{bmatrix} \tag{3}$$

2.2. 3D Vehicle Velocity

The vehicle’s translational velocity vector \dot{r}_0^v in $X_0Y_0Z_0$ is related to its body-fixed velocities in $X_vY_vZ_v$ as

$$\dot{r}_0^v = R_0^v(\bar{q}^v)v^v, \quad v^v \triangleq \dot{r}_v^v. \tag{4}$$

The relationship between the angular velocity vector of the vehicle in $X_vY_vZ_v$, $\omega^v = [\omega_x^v \ \omega_y^v \ \omega_z^v]^\top$ and the time derivative of the quaternion vector, $\dot{\bar{q}}^v$, is given by

$$\dot{\bar{q}}^v = \frac{1}{2}G^{v\top}(\bar{q}^v)\omega^v, \quad G^v(\bar{q}^v) = [-q^v, \ q_0^v I_3 - S(q^v)] \tag{5}$$

The time derivatives of the quaternions are derived from the unit vector constraint as $\bar{q}^{v\top} \dot{\bar{q}}^v = 0$.

2.3. 3D Vehicle State Equations

Using the vehicle pose and velocity equations, the equations of motion of the 3D vehicle are derived as

$$\dot{v}^v = -S(\omega^v)v^v + \frac{F^v}{m^v} \tag{6}$$

$$\dot{\omega}^v = -(I^v)^{-1}S(\omega^v)I^v\omega^v + (I^v)^{-1}\tau^v \tag{7}$$

where $F^v = [0 \ 0 \ F_z^v]^\top$ and $\tau^v = [\tau_x^v \ \tau_y^v \ \tau_z^v]^\top$ are the input force and moment vectors, respectively, m^v is the mass, and I^v is the 3×3 moment of inertia matrix of the vehicle. Given these equations, the 13×1 state and 6×1 input vectors are defined as

$$x^v = \left[r_0^{v\top} \ \bar{q}^{v\top} \ v^{v\top} \ \omega^{v\top} \right]^\top, \quad u = \left[F^{v\top} \ \tau^{v\top} \right]^\top \tag{8}$$

The general nonlinear state equations are then given as

$$\dot{x} = f^v(x^v) + g^v u \tag{9}$$

where

$$f^v(x^v) = \begin{bmatrix} R_0^v(\bar{q}^v)v^v \\ \frac{1}{2}G^{v\top}(\bar{q}^v)\omega^v \\ -S(\omega^v)v^v \\ -I^{v^{-1}}S(\omega^v)I^v\omega^v \end{bmatrix}, \quad g^v = \begin{bmatrix} 0_3 & 0_3 \\ 0_{4 \times 3} & 0_{4 \times 3} \\ \frac{1}{m^v}I_3 & 0_3 \\ 0_3 & I^{v^{-1}} \end{bmatrix} \tag{10}$$

where 0_3 and $0_{4 \times 3}$ are the 3×3 and 4×3 zero matrices, respectively. Linearization is required for any EKF base-state estimation. The 13×13 Jacobian matrix J^v of the state Equations (8) is derived as

$$J^v = \frac{\partial(f^v(x^v))}{\partial x^v} = \begin{bmatrix} 0_3 & \frac{\partial \dot{r}_0^v}{\partial \bar{q}^v} & R_0^v(\bar{q}^v) & 0_3 \\ 0_{4 \times 3} & \frac{\partial \dot{\bar{q}}^v}{\partial \bar{q}^v} & 0_{4 \times 3} & \frac{1}{2}G^{v\top}(\bar{q}^v) \\ 0_3 & 0 & 0_3 & -S(\omega^v) \\ 0_3 & 0 & 0_3 & 0_3 \end{bmatrix} \begin{bmatrix} S(v^v) \\ J_\omega \end{bmatrix} \tag{11}$$

where $J_\omega = I^{v^{-1}}[S(I^v\omega^v) - S(\omega^v)I^v]$, and $R_0^v(\bar{q}^v)$ and $G^v(\bar{q}^v)$ are from (2) and (5), respectively. Given that $\frac{\partial \dot{r}_0^v}{\partial \bar{q}^v} = \left[\frac{\partial \dot{r}_0^v}{\partial q_0^v} \ \frac{\partial \dot{r}_0^v}{\partial q^v} \right]$, the partial derivatives within the Jacobian matrix are derived as

$$\frac{\partial \dot{r}_0^v}{\partial q_0^v} = \frac{\partial Rv^v}{\partial q_0^v} = [(4q_0^v)I_3 + 2S(q^v)]v^v \tag{12}$$

$$\frac{\partial \dot{r}_0^v}{\partial q^v} = \frac{\partial Rv^v}{\partial q^v} = 2q^v v^{v\top} + 2(q^{v\top} v^v)I_3 - 2q_0^v S(v^v) \tag{13}$$

$$\frac{\partial \dot{q}^v}{\partial \bar{q}^v} = \frac{1}{2} \begin{bmatrix} 0 & -\omega^{v\top} \\ \omega^v & -S(\omega^v) \end{bmatrix} \tag{14}$$

2.4. Ground Vehicle Model

While our focus in this work is on 3D vehicles, many applications involve networks of ground and air vehicles. Hence, in this section, a model ground vehicles is presented. Figure 1 shows the top view of a ground vehicle’s pose with respect to the inertial reference frame. The pose of a mobile robot is given by its position (x, y, z) and its yaw angle ϕ . Note that the z component is added since the ground may not be horizontal, and quaternions are not necessary due to a single rotational degree of freedom. Hence, the 7×1 state and 2×1 input vectors for the ground vehicle are defined as

$$x^v = \begin{bmatrix} r_0^v \\ \phi^v \\ v_x^v \\ z^v \\ \omega_z^v \end{bmatrix}, \quad u = \begin{bmatrix} F^v \\ \tau^v \end{bmatrix} \tag{15}$$

where $F^v = F_x^v$ and $\tau^v = \tau_z^v$. The components of the state Equations (10) for this ground vehicle model are given as

$$f^v(x^v) = \begin{bmatrix} v_x^v \cos(\phi^v) - d^v \omega_z^v \sin(\phi^v) \\ v_x^v \sin(\phi^v) + d^v \omega_z^v \cos(\phi^v) \\ 0 \\ \omega_z^v \\ d^v (\omega_z^v)^2 \\ 0 \\ -\frac{m^v d^v}{I_{zz}^v} \omega_z^v v_x^v \end{bmatrix}, \quad g^v = \begin{bmatrix} 0_{4 \times 1} & 0_{4 \times 1} \\ 1/m^v & 0 \\ 0 & 0 \\ 0 & 1/I_{zz}^v \end{bmatrix} \tag{16}$$

where d is the distance from the rear axle to the vehicle origin. Note that the lateral velocity component is derived based on the mobile robot configuration as $v_y^v = d\omega_z^v$ and therefore not included as a state. The Jacobian matrix of the linearized state equations is

$$\frac{\partial(f^v(x^v))}{\partial x^v} = \begin{bmatrix} 0 & 0 & 0 & -v_x^v \sin(\phi^v) - d^v \omega_z^v \cos(\phi^v) & \cos(\phi^v) & 0 & -d^v \sin(\phi^v) \\ 0 & 0 & 0 & v_x^v \cos(\phi^v) - d^v \omega_z^v \sin(\phi^v) & \sin(\phi^v) & 0 & d^v \cos(\phi^v) \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 2d^v \omega_z^v \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -\frac{m^v d^v}{I_{zz}^v} \omega_z^v & 0 & -\frac{m^v d^v}{I_{zz}^v} v_x^v \end{bmatrix} \tag{17}$$

3. Nonlinear Control Design

Since control design for ground vehicles is trivial and not our focus, we refer the reader to numerous methods, including [35]. Though more recent controllers are available [36], we present a summary of our nonlinear control design for quadcopters [37], for which the control law is derived using Euler angles, noting that conversion of Euler angles to quaternions through the orthogonal rotation matrix is trivial. In this section, we drop the superscript “ v ” to simplify the notation.

Consider the position vector r_0^v and Euler angle vector $\Phi = [\phi, \theta, \psi]^T$, where ϕ , θ , and ψ are the roll, pitch, and yaw angles, respectively. We will treat the quadcopter as a four–input–four–output system with the outputs being r_0^v and ψ and the inputs being the total thrust force $F_z^v = f$ and the three control moments $\tau \equiv \tau^v$. In order to derive the nonlinear control law, we divide the system into three subsystems. The first subsystem consists of the roll and pitch moments τ_x, τ_y as the inputs and the x, y components of r_0^v as the outputs. The second subsystem has f as the input and the z component of r_0^v as the output, and the third subsystem has yaw moment τ_z as the input and yaw angle ψ as the output.

3.1. Roll and Pitch Moments

The first subsystem has a relative degree 4, i.e., one must take up to the fourth time derivatives of the outputs x, y to reach the inputs τ_x, τ_y . To do this, we use the system model for the translational part of the equations of motion (6) in the earth-fixed frame, which may be written as:

$$m\ddot{r} = R \begin{bmatrix} 0 \\ 0 \\ \hat{f} \end{bmatrix} - \begin{bmatrix} 0 \\ 0 \\ mg \end{bmatrix} \tag{18}$$

where $R \equiv R_0^v$, g is acceleration due to gravity, and we have replaced the thrust force f with its nominal value \hat{f} . Next, we take the first and second time derivatives of Equation (18):

$$m\ddot{\dot{r}} = R \left(\begin{bmatrix} 0 \\ 0 \\ \dot{\hat{f}} \end{bmatrix} + \tilde{\omega} \begin{bmatrix} 0 \\ 0 \\ \hat{f} \end{bmatrix} \right) \tag{19}$$

$$m\dot{r}^{(4)} = R \left(\begin{bmatrix} 0 \\ 0 \\ \ddot{\hat{f}} \end{bmatrix} + 2\tilde{\omega} \begin{bmatrix} 0 \\ 0 \\ \dot{\hat{f}} \end{bmatrix} + (\dot{\tilde{\omega}} + \tilde{\omega}\tilde{\omega}) \begin{bmatrix} 0 \\ 0 \\ \hat{f} \end{bmatrix} \right) \tag{20}$$

where $\ddot{\dot{r}}$ and $r^{(4)}$ are called jerk and snap, respectively. Equation (20) directly relates τ_x and τ_y to snap by substituting for $\dot{\omega}_x$ and $\dot{\omega}_y$ from (7). Note that $\dot{\omega}_z$ and consequently τ_z do not appear in (20).

Next, we define a smooth reference trajectory $r_r = [x_r(t), y_r(t), z_r(t)]^T$ and the error vector $e_r = r - r_r$ and propose a control law that ensures fourth-order exponentially stable error dynamics with an integral term:

$$e_r^{(4)} + 5\lambda\ddot{e}_r + 10\lambda^2\dot{e}_r + 10\lambda^3e_r + 5\lambda^4e_r + \lambda^5 \int_0^t e_r(\sigma)d\sigma = 0 \tag{21}$$

where $\dot{e}_r = \dot{r} - \dot{V}_r$, $\ddot{e}_r = \ddot{r} - \ddot{V}_r$, $e_r^{(4)} = r^{(4)} - \ddot{V}_r$, and the integral term is added to reject constant and slowly changing biased (wind) disturbances. For simplicity, the coefficients in the above error dynamics are selected such that all the eigenvalues are at $-\lambda, \lambda > 0$, i.e., the fastest response without oscillations predicted by the model.

Premultiplying Equation (21) by mR^T , we may write the equation in the following form:

$$mR^T\ddot{\dot{e}}_r + 5\lambda mR^T\dot{e}_r + 10\lambda^2 mR^T e_r + \gamma = 0 \tag{22}$$

where

$$\gamma = mR^T (-\ddot{V}_r - 5\lambda\dot{V}_r - 10\lambda^2\dot{V}_r + 10\lambda^3\dot{e}_r + 5\lambda^4e_r + \lambda^5 \int_0^t e_r(\sigma)d\sigma) \tag{23}$$

We then substitute from Equations (18)–(20) into Equation (22):

$$\begin{bmatrix} \hat{f}\dot{\omega}_y \\ -\hat{f}\dot{\omega}_x \\ \ddot{\hat{f}} \end{bmatrix} = \begin{bmatrix} -2\omega_y \\ 2\omega_x \\ -5\lambda \end{bmatrix} \dot{\hat{f}} - \begin{bmatrix} 5\lambda\omega_y + \omega_z\omega_x \\ -5\lambda\omega_x + \omega_y\omega_z \\ 10\lambda^2 - \omega_x^2 - \omega_y^2 \end{bmatrix} \hat{f} + 10\lambda^2mg \begin{bmatrix} R_{31} \\ R_{32} \\ R_{33} \end{bmatrix} - \gamma \tag{24}$$

where R_{31} , R_{32} , and R_{33} are the corresponding elements of the rotation matrix R .

Substituting for $\dot{\omega}_x$ and $\dot{\omega}_y$ from the first two equations of (7) into the first two equations in (24) yields the roll and pitch control moments:

$$\tau_x = (I_{xx} - I_{yy} + I_{zz})\omega_y\omega_z - 5\lambda I_{xx}\omega_x - \frac{I_{xx}}{\hat{f}} [2\omega_x\dot{\hat{f}} + 10\lambda^2mgR_{32} - \gamma_2] \tag{25}$$

$$\tau_y = (I_{xx} - I_{yy} - I_{zz})\omega_z\omega_x - 5\lambda I_{yy}\omega_y + \frac{I_{yy}}{\hat{f}} [-2\omega_y\dot{\hat{f}} + 10\lambda^2mgR_{31} - \gamma_1] \tag{26}$$

where γ_1, γ_2 are the corresponding elements of vector γ in (23), and \hat{f} and $\dot{\hat{f}}$ are derived by integrating $\ddot{\hat{f}}$; from the third row of Equation (24) with the hovering/rest initial conditions $\hat{f}(0) = mg$ and $\dot{\hat{f}}(0) = 0$:

$$\ddot{\hat{f}} = -5\lambda\dot{\hat{f}} - (10\lambda^2 - \omega_x^2 - \omega_y^2)\hat{f} + 10\lambda^2mgR_{33} - \gamma_3$$

It must be noted that the system model has allowed us to compute the control laws in (25) and (26) in terms of the system states only: i.e., pose variable (r, Φ) and body velocities (v, ω) . Thus, estimates of higher-order derivatives such as acceleration, jerk, and snap are not required. Only the reference trajectory requires higher-order derivatives and thus must be defined in terms of sufficiently smooth time functions.

3.2. Thrust Force

The thrust control force f appears in the third translational equation of motion and thus has a relative degree of 2. Hence, we define second-order exponentially stable error dynamics with an integral term and both eigenvalues at $-\lambda_z, \lambda_z > 0$:

$$\ddot{e}_Z + 3\lambda_z\dot{e}_Z + 3\lambda_z^2e_Z + \lambda_z^3 \int_0^t e_Z(\sigma)d\sigma = 0 \tag{27}$$

where $e_Z = z - z_r(t)$, $\dot{e}_Z = \dot{z} - \dot{z}_r(t)$, and $\ddot{e}_Z = \ddot{z} - \ddot{z}_r(t)$. The third equation in (18) yields $\ddot{z} = R_{33}f/m - g$. Thus, the thrust force may be derived using Equation (27):

$$f = \frac{m}{R_{33}} (g + \ddot{z}_r - 3\lambda_z\dot{e}_Z - 3\lambda_z^2e_Z - \lambda_z^3 \int_0^t e_Z(\sigma)d\sigma). \tag{28}$$

3.3. Yaw Control

Since τ_z can be expressed in terms of the second time derivatives of the yaw angle output, we again define second-order exponentially stable error dynamics with an integral term and both eigenvalues at $-\lambda_\psi, \lambda_\psi > 0$:

$$\ddot{e}_\psi + 3\lambda_\psi \dot{e}_\psi + 3\lambda_\psi^2 e_\psi + \lambda_\psi^3 \int_0^t e_\psi(\sigma) d\sigma = 0 \tag{29}$$

where $e_\psi = \psi - \psi_r(t), \dot{e}_\psi = \dot{\psi} - \dot{\psi}_r(t), \ddot{e}_\psi = \ddot{\psi} - \ddot{\psi}_r(t)$, and $\psi_r(t)$ represents the continuously differentiable reference yaw trajectory. We then define the kinematic relationship between angular acceleration $\dot{\omega}_z$ and the Euler yaw acceleration $\ddot{\psi}$ as [37]:

$$\ddot{\psi} = [\tau_z - (I_{yy} - I_{xx})\omega_x\omega_y] / I_{zz} - \sin\theta\ddot{\phi} - \cos\theta\dot{\theta}\dot{\psi}. \tag{30}$$

The yaw control moment may then be derived by substituting from (30) into (29):

$$\tau_z = (I_{yy} - I_{xx})\omega_x\omega_y + I_{zz}[\sin\theta\ddot{\phi} + \cos\theta\dot{\theta}\dot{\psi} + \gamma_\phi] \tag{31}$$

where $\gamma_\phi = \ddot{\psi}_r(t) - 3\lambda_\psi \dot{e}_\psi - 3\lambda_\psi^2 e_\psi - \lambda_\psi^3 \int_0^t e_\psi(\sigma) d\sigma$.

3.4. Relation between Control and Rotor Inputs

The quadrotor motor inputs are expressed in terms of pulse width modulation (PWM) values. Hence, we must relate the four control inputs, total thrust force f , and the three control moments τ to the motor PWM values. This is done in a two-step process. First, consider the top view of the four rotor arrangements with rotor speeds $\Omega_i, i = 1, 2, 3, 4$, as shown in Figure 2a. The control inputs are related to the four input rotor speeds via the following invertible transformation:

$$\begin{bmatrix} f \\ \tau_x \\ \tau_y \\ \tau_z \end{bmatrix} = \begin{bmatrix} c_T & c_T & c_T & c_T \\ dc_T & -dc_T & -d_3c_T & dc_T \\ dc_T & -dc_T & dc_T & -dc_T \\ -c_Q & -c_Q & c_Q & c_Q \end{bmatrix} \begin{bmatrix} \Omega_1^2 \\ \Omega_2^2 \\ \Omega_3^2 \\ \Omega_4^2 \end{bmatrix} \tag{32}$$

where d is the arm length and $c_T, c_Q > 0$ are the experimentally derived equivalent thrust and torque coefficients. A series of experiments were then performed by applying PWM values and measuring the resulting rotor speeds with a laser tachometer. The resulting data along with the best-fit plot are shown in Figure 2b, and the relationship is given by:

$$PWM = 0.0012\Omega^2 + 0.0684\Omega + 1132.59 \tag{33}$$

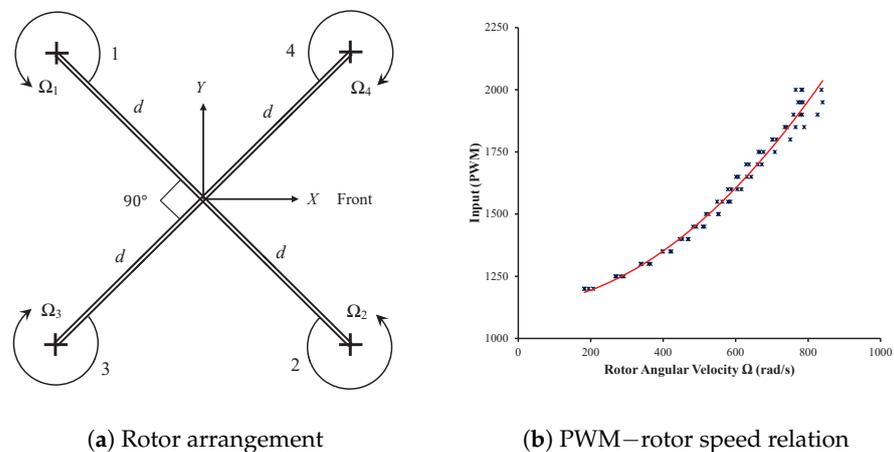


Figure 2. (a) Top view of the quadrotor model; (b) PWM vs. rotor speed and its best fit model in red.

4. Multi-Vehicle System Model

The controller presented in the previous section requires state feedback. However, it may be essential not to rely on absolute position feedback from GPS in multi-vehicle systems. Here, we present the case of multi-vehicle systems for which relative position measurements are employed to formulate the cooperative localization algorithm for improved state estimation. We assume that each vehicle possesses an IMU to measure its own orientation fairly accurately, which is a reasonable assumption.

Consider the multi-vehicle system shown in Figure 3. In this system, the base vehicle b can measure the position of the target vehicle t relative to its own body-fixed reference frame. We denote the relative position measurement vector by $z = r$, which is given in terms of the system states as:

$$r = R_0^{b\top} r_0^{bt} = R_0^{b\top} (r_0^t - r_0^b) = R_0^{b\top} r_0 \tag{34}$$

where $r_0 = r_0^t - r_0^b$.

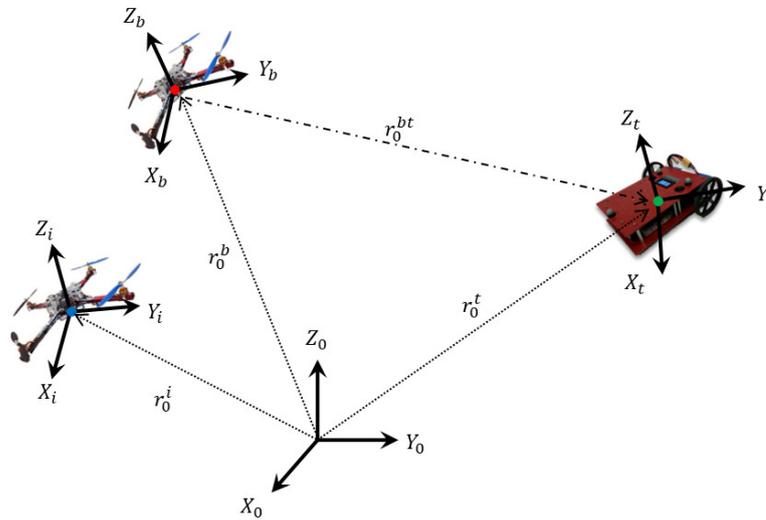


Figure 3. Absolute and relative pose geometry.

The linearized measurement equation is given by $z = Hx$, where $H = [H^b \ H^t]$. The structure and size of this matrix depends on the types of vehicles involved in the measurement. If both the base and target are 3D vehicles, the 3×26 H matrix is given as:

$$H = \begin{bmatrix} -R_0^{b\top} & \frac{dr}{dq^b} & 0_{3 \times 6} & R_0^{b\top} & 0_{3 \times 10} \end{bmatrix} \tag{35}$$

and $\frac{dr}{dq^b} = \begin{bmatrix} \frac{dr}{dq_0^b} & \frac{dr}{dq^b} \end{bmatrix}$ can be determined using (12):

$$\frac{dr}{dq_0^b} = [(4q_0^b)I_3 + 2S(q^b)]r_0 \tag{36}$$

$$\frac{dr}{dq^b} = 2q^b r_0^\top + 2(q^{b\top} r_0)I_3 - 2q_0^b S(r_0). \tag{37}$$

As shown in Figure 3, a network may comprise 2D and 3D vehicles. Since the state vector for 2D vehicles is different, the linearized H matrix will have a different size and

components if the base or target is a ground vehicle. When a base ground vehicle measures the relative position of a target ground (target) vehicle, the 3×14 H matrix is defined as

$$H = \begin{bmatrix} -R_0^{b\top} & \frac{\partial r}{\partial \phi^b} & 0_{3 \times 3} & R_0^{b\top} & 0_{3 \times 4} \end{bmatrix} \tag{38}$$

where

$$\frac{\partial r}{\partial \phi^b} = \begin{bmatrix} -\sin \phi^b & \cos \phi^b & 0 \\ -\cos \phi^b & -\sin \phi^b & 0 \\ 0 & 0 & 0 \end{bmatrix} r_0 \tag{39}$$

In cases for which a base ground vehicle measures the relative position of a 3D vehicle, we have

$$H = \begin{bmatrix} -R_0^{b\top} & \frac{\partial r}{\partial \phi^b} & 0_{3 \times 3} & R_0^{b\top} & 0_{3 \times 10} \end{bmatrix}. \tag{40}$$

Finally, when a base air vehicle measures the relative position of a ground vehicle, we have

$$H = \begin{bmatrix} -R_0^{b\top} & \frac{\partial r}{\partial \phi^b} & 0_{3 \times 6} & R_0^{b\top} & 0_{3 \times 4} \end{bmatrix}. \tag{41}$$

5. Cooperative Localization

Let us consider a system of n vehicles numbered 1 through n . At sample time (k) , a vehicle i uses absolute measurements to propagate its pose using the noisy discrete form of (9) as follows:

$$\mathbf{x}^i(k+1) = f^i(\mathbf{x}^i(k)) + g^i(k)u^i(k) + \gamma^i(\mathbf{x}^i(k))\eta^i(k) \tag{42}$$

where \mathbf{x}^i , u^i , and η^i represent the state, input, and process noise vectors, respectively, and g^i and $\gamma^i(\mathbf{x}^i)$ are the input and process noise coefficient functions.

Now, let us assume that vehicle i in the set $\{1, \dots, n\}$ can take a relative measurement of the pose of a vehicle $j \neq i$. In that case, the measurement vector z^{ij} is given by

$$z^{ij}(k+1) = h^{ij}(\mathbf{x}^i(k), \mathbf{x}^j(k)) + v^i(k) \tag{43}$$

where h^{ij} is defined in (34), and $v^i(k)$ represents the measurement noise.

Assuming the initial conditions $\mathbf{x}^i(0)$ and starting with an initial guess $P^i(0) > 0$ with zero cross-covariance, vehicles $i = 1, \dots, n$ can predict their states by propagation as:

$$\begin{aligned} \hat{\mathbf{x}}^{i-}(k+1) &= f(\mathbf{x}^{i+}(k)) + g^i(k)u^i(k) \\ P^{i-}(k+1) &= J^i(k)P^{i+}(k)J^{i\top}(k) + \Gamma_i(k)Q^i\Gamma_i^\top(k), \\ P^{ij-}(k+1) &= J^i(k)P^{i+}(k)J^{j\top}(k), \quad j = 1, \dots, n, j \neq i \end{aligned} \tag{44}$$

where $J^i(k)$ is calculated using (11), $Q^i > 0$ is the covariance of the process noise, and $\Gamma^i = \frac{\partial \gamma^i}{\partial \mathbf{x}^i}$.

When a “base” vehicle b takes relative measurements of a “target” vehicle t , the innovation error is given by:

$$\begin{aligned} e^{bt}(k+1) &= z^{bt} - h^{bt}(\hat{\mathbf{x}}^{b-}(k+1), \hat{\mathbf{x}}^{t-}(k+1)) \\ &\approx z^{bt} - \begin{bmatrix} H^b(k+1) & H^t(k+1) \end{bmatrix} \begin{bmatrix} \hat{\mathbf{x}}^{b-}(k+1) \\ \hat{\mathbf{x}}^{t-}(k+1) \end{bmatrix} \end{aligned} \tag{45}$$

where H^b and H^t are defined in (35), (38), (40), or (41) depending on the vehicles involved in the relative measurement. The cross-covariance terms create a slight modification to the innovation covariance matrix:

$$S^{bt} = H^b P^{b-} H^{b\top} + H^t P^{t-} H^{t\top} + H^b P^{bt-} H^{t\top} + H^t P^{tb-} H^{b\top} + \mathcal{R}^{bt} \tag{46}$$

Using (45) and (46), the Kalman gains, states, and covariances for all vehicles $i = 1, \dots, n$ are updated as:

$$\begin{aligned} K^i &= [P^{it-} H^{t\top} + P^{ib-} H^{b\top}] S^{bt-1} \\ \hat{x}^{i-} &= \hat{x}^{i-} + K^i e^b \\ P^{i+} &= P^{i-} - K^i S^{bt} K^{i\top} \\ P^{ij+} &= P^{ij-} - K^i S^{bt} K^{j\top}, \quad j = 1, \dots, n, \quad j \neq i. \end{aligned} \tag{47}$$

As long as the measurement noise is reasonable and the EKF linearization is accurate, the semi-definite Kalman gain matrix can guarantee a reduction in localization error with every relative measurement. A block diagram of control system implementation with the help of relative measurements and cooperative localization is shown in Figure 4.

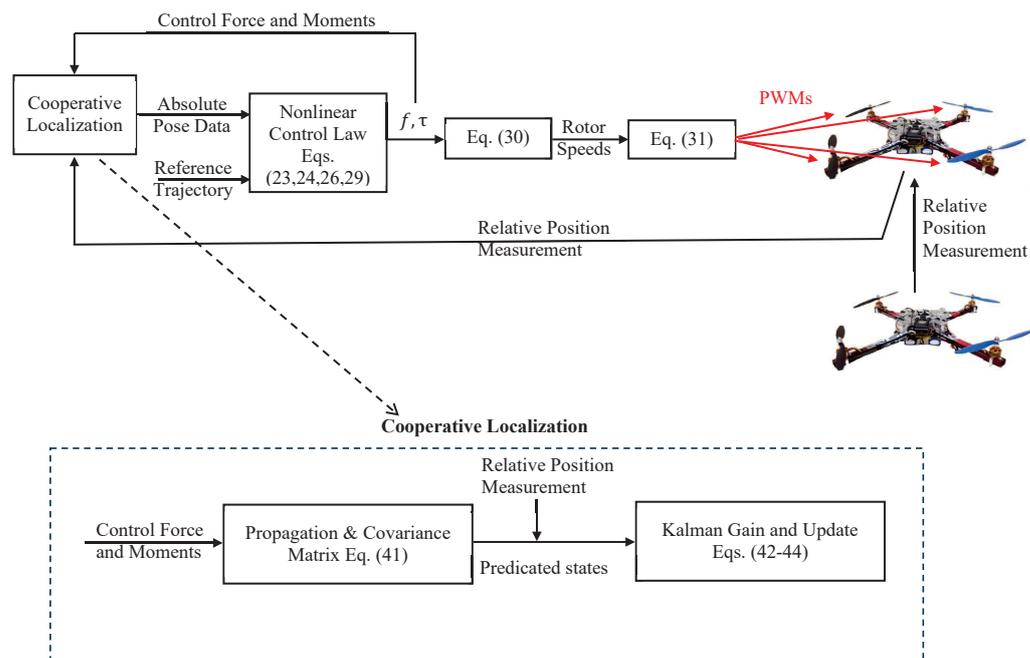


Figure 4. Block diagram of control system implementation with the help of cooperative localization.

6. Results

We have performed a variety of simulations and experiments to verify the CL algorithm’s effectiveness in improving state estimation and closed-loop control performance.

6.1. Experimental Setup

The experiments were performed in a 4.5 m × 4.0 m × 3.0 m cage with a state-of-the-art vision system capable of 3D localization with sub-millimeter accuracy at 250 frames per second. We formed heterogeneous vehicle networks with small in-house-designed quadcopters and wheeled mobile robots representing ground vehicles.

Figure 5 shows details of one of the quadcopters and mobile robots. The quadcopter components include DJI Flame Wheel Kits with ESC (electronic speed control) motors

and propellers, a Raspberry Pi 3 B+ for processing and IO purposes, a PWM driver for the four ESC motors, an OLED display, a 3000 mAh lithium–polymer battery, a voltage measurement unit, and a simple remote relay. They are also equipped with a custom-designed PCB board to simplify the driver and sensor connections and an IMU. The mobile robots are differential-drive with two motors, a dual-motor control, a Raspberry Pi 3 B+, an Arduino Due board, an IMU, and a LiDAR sensor. All vehicles have infrared markers so that their motion can be accurately tracked by the motion-capture system and reported for comparison purposes.

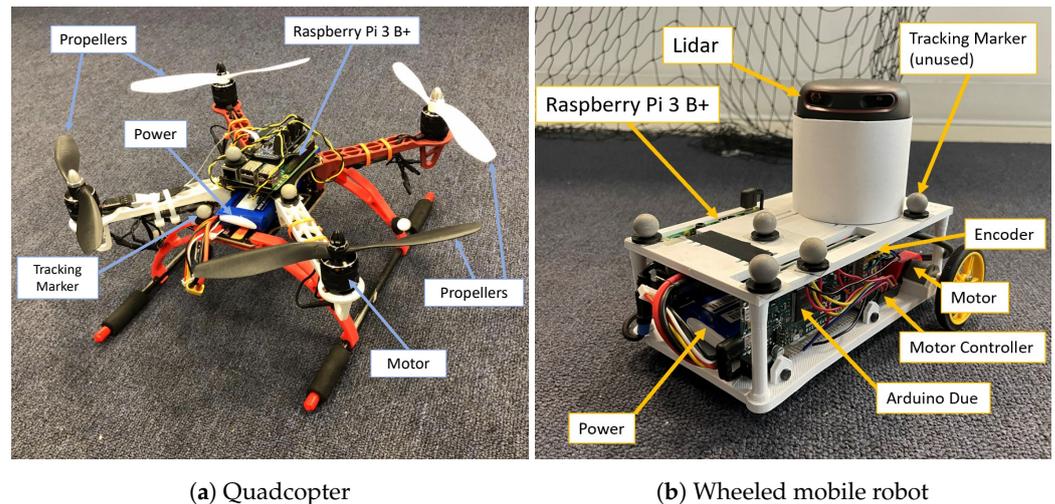


Figure 5. The experimental in-house quadcopter (a) and ground vehicle (b).

For quadcopters, the control software calculates the three moments and thrust force according to (25)–(28). It then calculates the corresponding rotor speeds and converts the data to PWM motor commands by interpolating a lookup table [37]. For mobile robots, the controller outputs calculate the surge force and yaw moment and then converts them to wheel speeds. These speeds are sent from the Raspberry Pi to an Arduino Due. A fast-rate PI closed-loop control then maintains the commanded wheel speeds for the mobile robot.

We used identical quadcopters and identical mobile robots in our experiments. The mass and geometric properties, which are primarily used for state estimation in CL, are:

$$\begin{aligned}
 m &= 1.01 \text{ kg}, & I &= \text{diag}(0.0066, 0.0065, 0.0109) \text{ kg m}^2 \\
 d &= 0.226 \text{ m}, & c_T &= 7 \times 10^{-6} \text{ kgm} & c_Q &= 2.6 \times 10^{-7} \text{ kg m}^2 & \text{quadcopters} \\
 m &= 1.748 \text{ kg}, & I_{zz} &= 0.0126 \text{ kg m}^2, & d &= 0.05 \text{ m} & \text{mobile robots}
 \end{aligned}$$

The measurements were all take at 50 Hz, i.e., $\Delta t = 0.02 \text{ s}$. Meanwhile, the same rate was selected for propagation of the state equations for CL. We assumed a standard deviation of 0.005 m for all position-state measurements and 0.1° for all orientation-state measurements. We also assumed the process noise covariance matrix to be diagonal, with 0.1 values as diagonal elements.

6.2. Validation Results

We initially conducted a series of experiment designed to validate the CL algorithm. In these experiments, the closed-loop control algorithm was implemented without the use of the CL algorithm in real time. Instead, the control signals were collected and applied as inputs to the vehicle simulation models with CL. The simulation results were then compared with the recorded motion-capture system data to evaluate the CL algorithm performance.

6.2.1. Results with Distance-Based Measurements

We initially considered a three-vehicle network consisting of two quadcopters (Vehicles 1 and 2) and a mobile robot (Vehicle 3). All three vehicles were directed to follow predetermined circular paths using the control algorithm, as shown in Figure 6c. In order to test the CL algorithm, we assumed Vehicle 3 has continuous access to its own absolute position and that relative measurements depend on the proximity of the vehicles. Specifically, the quadcopters (Vehicles 1 and 2) can measure their relative positions when within 4 m of each other and the relative position of the mobile robot (Vehicle 3) when it is within a 2 m distance.

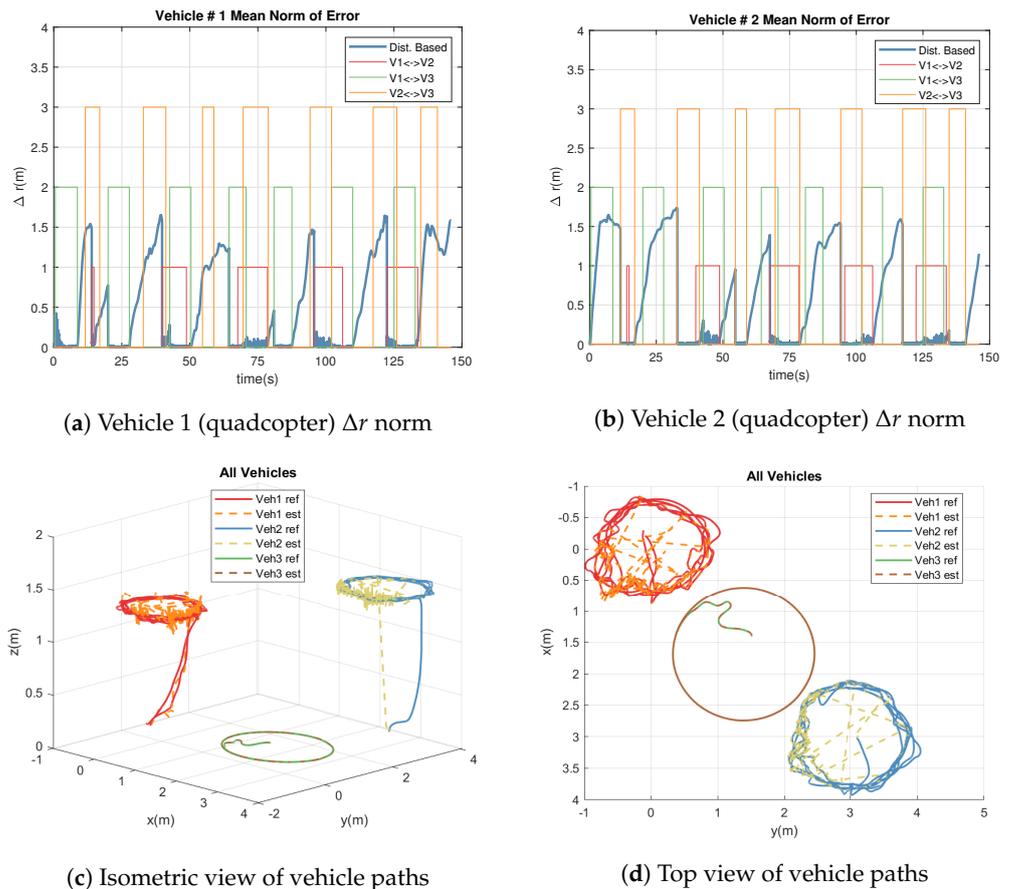


Figure 6. Three-vehicle test case with distance-based relative measurements: error norm and measurement flags for Vehicle 1 (a); error norm and measurement flags for Vehicle 2 (b); estimated versus actual paths: isometric view (c) and top view (d).

The estimation error results are shown in Figure 6a,b in the form of the time history of the position-error norms for Vehicles 1 and 2. The estimation error for Vehicle 3 stays relatively small due to consistent availability of the absolute position measurements and therefore is uneventful and is not shown. In the figures, the actual error is identified as “Dist. Based”. Three flags of magnitudes 1, 2, and 3, are used to distinguish when relative measurements between Vehicles 1 and 2, 1 and 3, and 2 and 3, respectively, are active. The figure illustrates that when the distance between the mobile robot and each quadcopter is within the threshold (i.e., Flags 1 and 2), there is a sharp drop in the position error of the quadcopters. In addition, the relative measurements between the two quadcopters can significantly improve localization if one has access to Vehicle 3. For instance, in Figure 6b at 40 s, Vehicle 2 does not have access to Vehicle 3, but its estimation error drops near zero due to relative measurement between Vehicles 1 and 2 (red line) and Vehicles 1 and 3 (green line). The isometric and top views of the reference (ref) and CL-estimated (est) paths of the three vehicles are presented in Figure 6c,d. These figures demonstrate significant variations

to quadcopter localization whenever they do not have any direct or indirect access to the mobile robot (Vehicle 3), which has access to absolute measurements.

Next, we consider the same scenario only now with the mobile robot having intermittent access to its absolute position (simulating occasionally available GPS). We assume continuous “GPS” access for Vehicle 1 for only 20s intervals starting at 30 s, 70 s, and 110 s. This intermittent access is a more realistic representation and shows how this sparsity will affect the overall network localization. Figure 7a–c show the position error norms for Vehicles 1–3, respectively, with shaded areas representing access to GPS. We can make the same conclusions as for the previous case, indicating that even when GPS is sparsely available, it is still beneficial to the vehicles in the network. For example, this can be observed in sharp error drops for Vehicle 2’s position error in Figure 7b at about 30 s, 70 s, 90 s, and 120 s when there are relative measurements of Vehicle 3 available and Vehicle 3 has access to GPS. However, another important conclusion is the significance of relative measurements for absolute localization in the absence of any absolute measurements. For instance, Figure 7a–c show that the localization errors are significantly reduced at 15 s and 90 s for all three vehicles due to relative measurements only since there is no access to GPS by Vehicle 1.

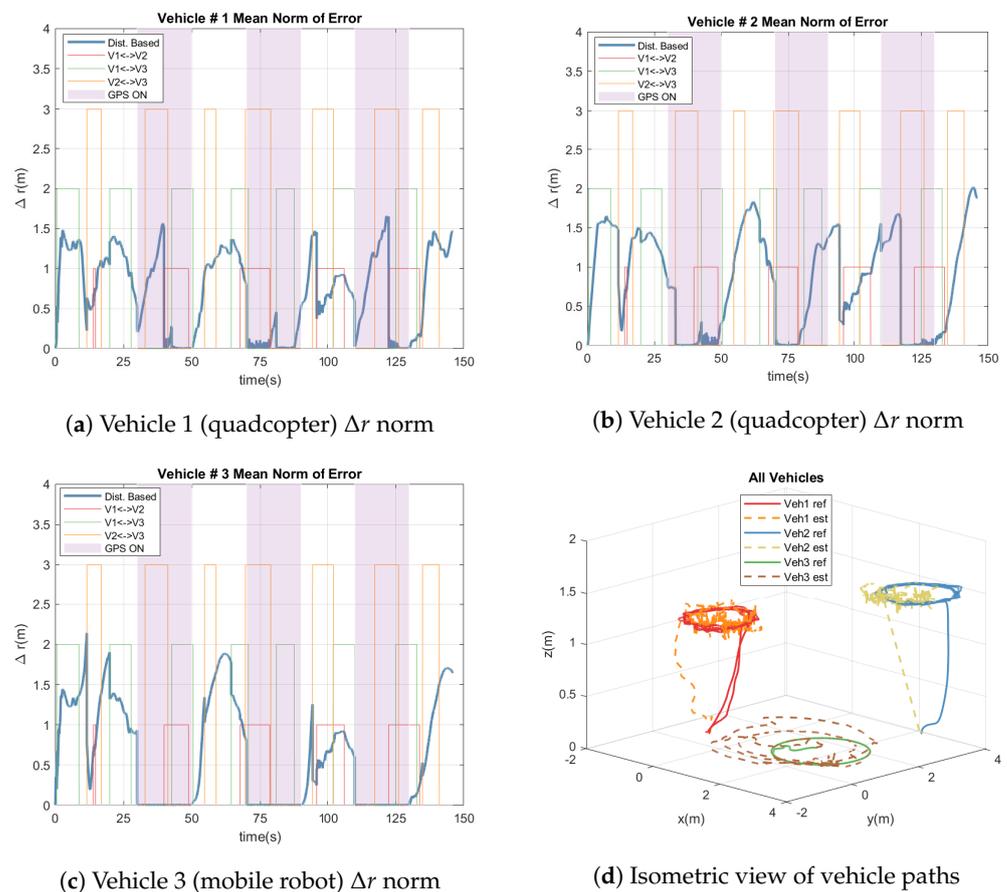


Figure 7. Three–vehicle test case with distance–based relative measurements and intermittent Vehicle 1 GPS access: error norm and measurement flags for Vehicle 1 (a); error norm and measurement flags for Vehicle 2 (b); estimated versus actual paths: isometric view (c) and top view (d).

6.2.2. Double-Caravan Measurement Formation

Next, we experimented with a much larger network of eight vehicles in a double-caravan measurement configuration. Each caravan was structured with a “leader” quadcopter vehicle and three mobile robot “followers” such that each vehicle can measure the relative position of its lead vehicle. We numbered the vehicles such that Vehicles 1–3 and 5–7 are mobile robots and Vehicles 4 and 8 are quadcopters. In this scenario, we assume

Vehicles 1, 2, 3, 5, 6, and 7 can only measure the relative positions of Vehicles 2, 3, 4, 6, 7, and 8, respectively. Meanwhile, the lead quadcopters (i.e., Vehicles 4 and 8) may or may not have access to GPS. Similar to the previous network, all eight vehicles were directed to follow predetermined circular paths using the control algorithm.

To facilitate information exchange across caravan formations, a specific role is assigned to the two follower Vehicles 1 and 5 (mobile robots) and next to the two lead Vehicles 4 and 8 (quadcopters). These designated “communication” vehicles possess the capacity to access and retrieve the relative position data of one another. This strategic assignment serves as the linchpin for the seamless propagation of cross-covariance and localization information across all vehicles within the caravans. Notably, this operational paradigm is achieved despite communication being primarily limited to the pairing of just two vehicles from each respective caravan formation.

We consider four slightly different measurement cases for this formation, listed in Table 1, to evaluate the CL algorithm’s performance. In Cases 1 through 3, Vehicle 1 measures the relative position of Vehicle 5 of the second caravan. In Case 1, there are only relative measurements. In Cases 2 and 3, in addition to the relative measurements, the lead quadcopter of the first caravan, Vehicle 4, has continuous or intermittent absolute position measurement access, respectively. The intermittent access to the absolute position for Vehicle 4 is provided after the first 40s and for 40s duration and then alternates regularly. Finally, in Case 4, there are again only relative measurements, as only the lead quadcopter of the first caravan, Vehicle 4, can measure the relative position of the lead quadcopter, Vehicle 8, of the second caravan.

Table 1. Simultaneous measurement descriptions for double–caravan measurement cases.

Simultaneous Measurements ($b \rightarrow t$)	
Case 1	1 → 2, 2 → 3, 3 → 4, 5 → 6, 6 → 7, 7 → 8, 1 → 5
Case 2	1 → 2, 2 → 3, 3 → 4, 5 → 6, 6 → 7, 7 → 8, 1 → 5, 4 → 4 (continuous)
Case 3	1 → 2, 2 → 3, 3 → 4, 5 → 6, 6 → 7, 7 → 8, 1 → 5, 4 → 4 (intermittent)
Case 4	1 → 2, 2 → 3, 3 → 4, 5 → 6, 6 → 7, 7 → 8, 4 → 8

The position error norms for all vehicles are presented in Figure 8 for Vehicles 1–4 and Figure 9 for Vehicles 5–8. As expected, Case 2 provides the most promising results, with little estimation errors since one vehicle in the network has continuous access to its absolute position and there is a measurement/communication path from the rest of the network to that vehicle. The errors for Cases 1 and 4 have a similar pattern of increasing with time since there are no absolute measurements. However, Case 1 performs slightly better throughout the simulations of all vehicles, indicating that communication and relative measurements between two follower vehicles may be slightly more beneficial than between lead vehicles. In addition, Case 4 consistently results in noisier estimations. Finally, Case 3 demonstrates that even intermittent access to absolute measurements (e.g., GPS) by one vehicle results in full recovery of estimation errors by all vehicles. This is the inherent advantage of CL and results from cross-covariance terms in the covariance matrix.

The isometric views of the reference (ref) and estimated (est) paths and estimated paths of all vehicles for Cases 1 through 4 are shown in Figure 10. It can be clearly seen that both continuous and intermittent access to absolute measurements (e.g., GPS) by one vehicle results in excellent estimation.

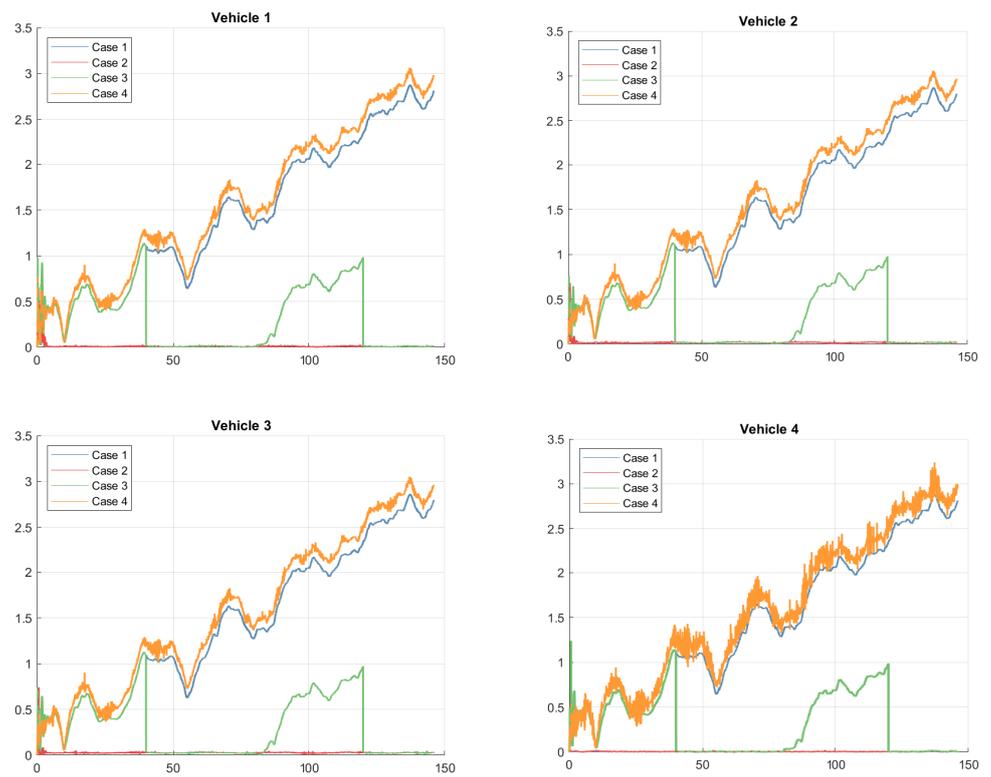


Figure 8. Position error norms of Vehicles 1–4 for the four measurement cases of double-caravan experiment.

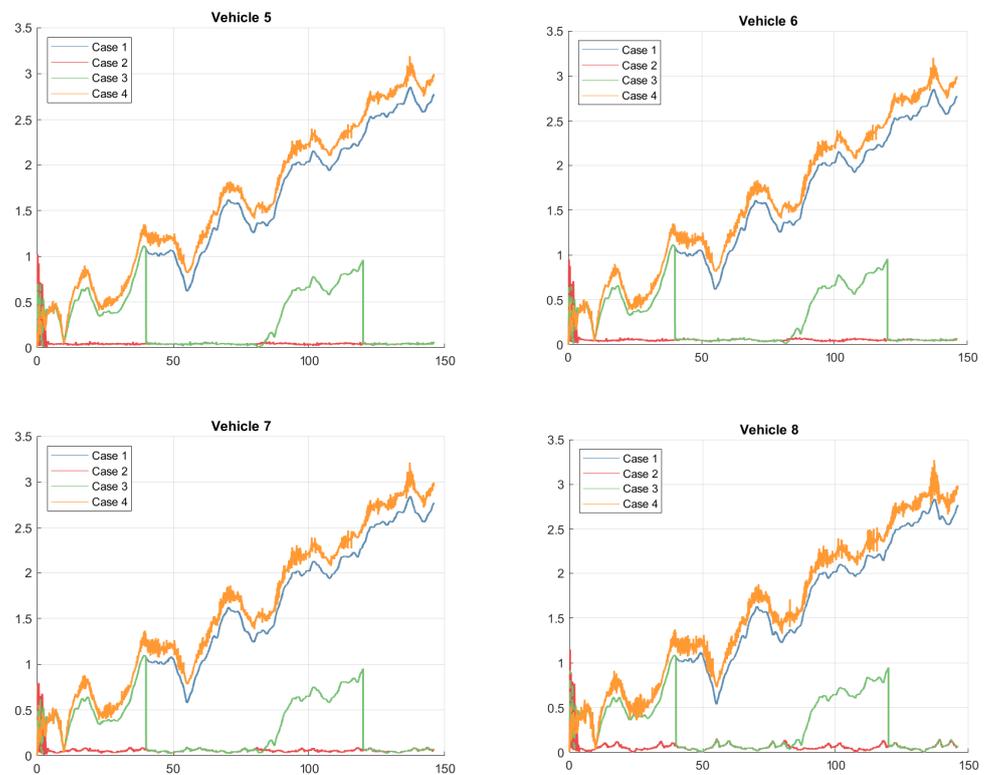


Figure 9. Position error norms of Vehicles 5–8 for the four measurement cases of double-caravan experiment.

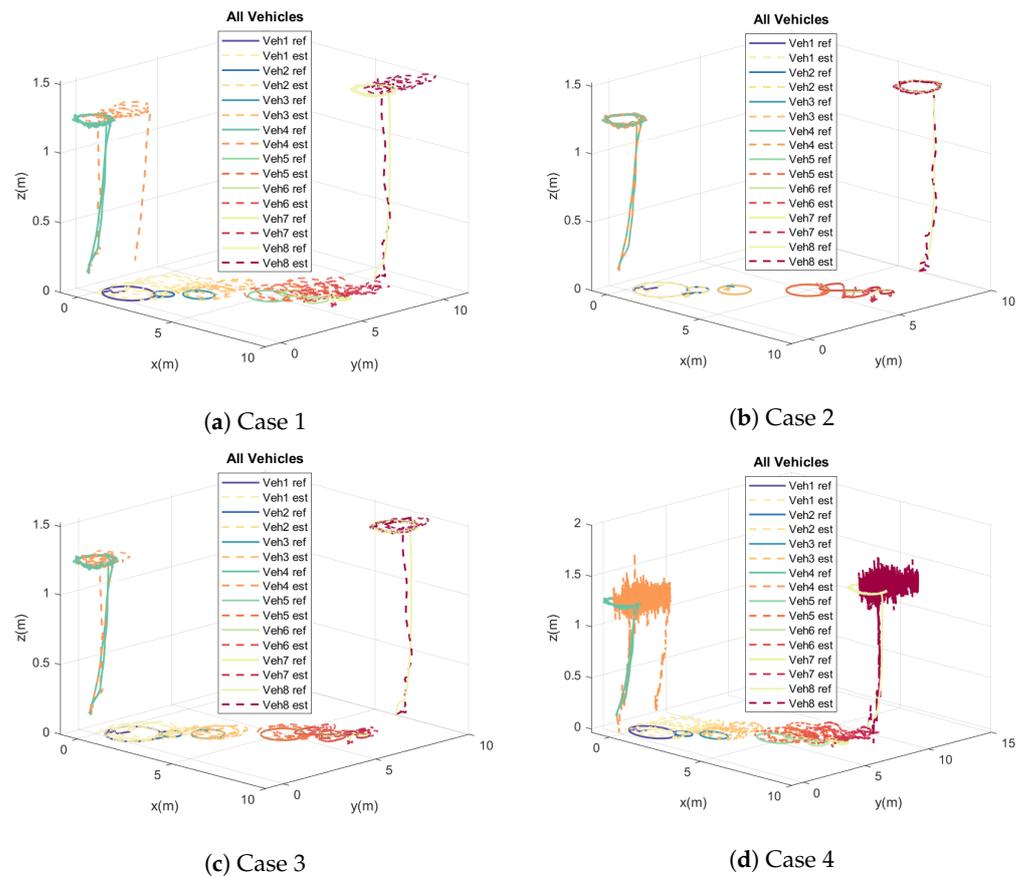


Figure 10. Estimated versus actual paths of all 8 vehicles for Cases 1 (a), 2 (b), 3 (c), and 4 (d) of the double-caravan experiment.

6.3. Closed-Loop Control Results

The final and most significant results of this work involve the application of the nonlinear closed-loop control presented in Section 3 to a network of three vehicles similar to Section 6.2 and a double-caravan network similar to Section 6.2.2.

6.3.1. Control of Three-Vehicle Network

First, we consider a three-vehicle network wherein all three vehicles are quadcopters. The vehicles initially lift off to a certain height and then follow predetermined circular paths using the closed-loop control algorithm presented in Section 3: once without and another time with the aid of the CL algorithm. In this scenario, Vehicle 1 is capable of continuously measuring its own absolute position, while Vehicles 2 and 3 exclusively receive relative position measurements from Vehicle 1.

Figure 11 illustrates the position error norms Δr and paths of Vehicles 1 through 3 to compare the performance of the closed-loop control algorithm with and without the aid of CL. The results are promising for both the closed-loop control algorithm and its integration with cooperative localization. There are two important observations. First, the use of CL in state estimation results in significantly reducing the noise in the response and thus significantly reduces the control effort to combat the resulting high-frequency oscillations. The second observation is that there is actually a slight increase in the error for the vehicle that has access to its own absolute measurements: i.e., Vehicle 1.

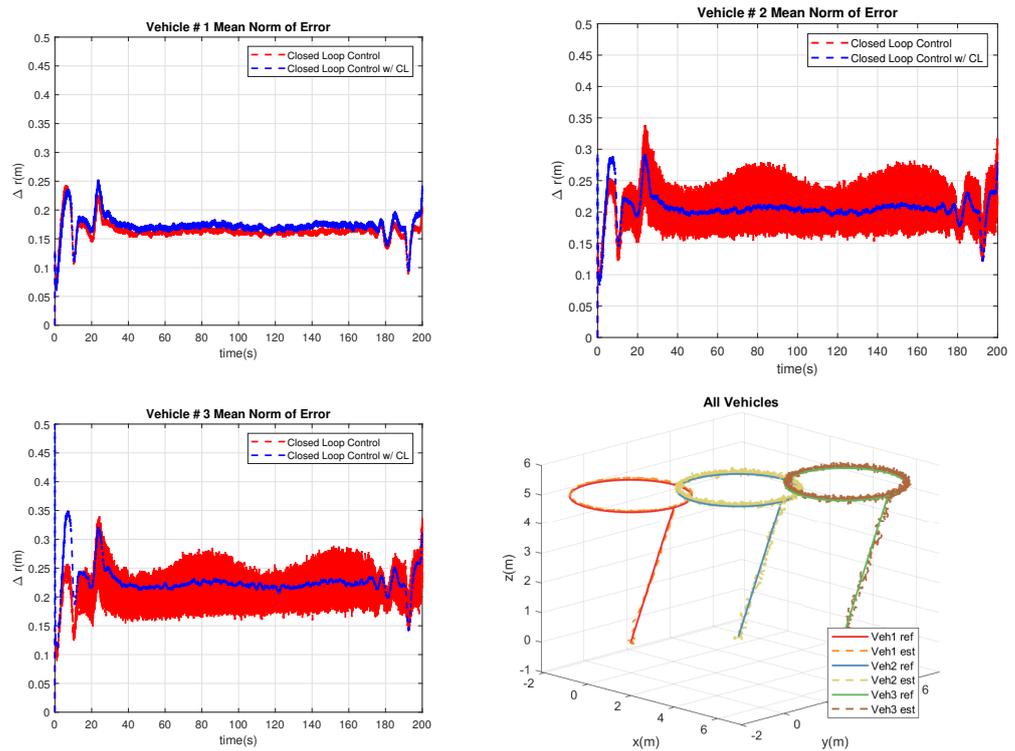


Figure 11. Comparison of the three-vehicle network position error norms and paths using closed-loop control with and without CL.

6.3.2. Control of Double-Caravan Formation

Consider the double-caravan arrangement as in Section 6.2.2 but with all eight vehicles being quadcopters. We consider two cases with the same relative measurement scenario as listed in Table 2. In other words, Vehicles 1, 2, 3, 5, 6, 7, and 8 measure the relative positions of Vehicles 2, 3, 4, 6, 7, 8, and 4, respectively. Meanwhile, the leader Vehicle 4 can either continuously measure its own absolute position (Case 1) or can do so intermittently (Case 2).

Table 2. Simultaneous measurement descriptions for double-caravan formation in closed-loop control.

Simultaneous Measurements ($b \rightarrow t$)	
Case 1	1 \rightarrow 2, 2 \rightarrow 3, 3 \rightarrow 4, 5 \rightarrow 6, 6 \rightarrow 7, 7 \rightarrow 8, 8 \rightarrow 4, 4 \rightarrow 4 (continuous)
Case 2	1 \rightarrow 2, 2 \rightarrow 3, 3 \rightarrow 4, 5 \rightarrow 6, 6 \rightarrow 7, 7 \rightarrow 8, 8 \rightarrow 4, 4 \rightarrow 4 (intermittent)

In these simulations, we explore two cases. In Case 1, Vehicle 4 receives persistent and uninterrupted access to GPS data. In contrast, Case 2 introduces intermittent GPS access for Vehicle 4 with predefined 40s intervals. These experimental modifications will help investigate the implications of consistent versus sporadic GPS availability on the overall system performance, specifically when nonlinear control is present.

Figures 12 and 13 present the time history of the position error norms for Vehicles 1–4 and for Vehicles 5–8, respectively, for Case 1. As in the previous network, Vehicle 4 being able to measure its own absolute position does not benefit from CL, and, in fact, its tracking error slightly increases. Moreover, similar to the previous network, cooperative localization significantly reduces the state estimation noise for all other vehicles. Another interesting phenomenon is that as we go farther away from Vehicle 4, the superiority of CL in reducing the estimation error becomes more apparent, such that Vehicles 1 and 5 see the most benefit, followed by vehicles 2 and 6.

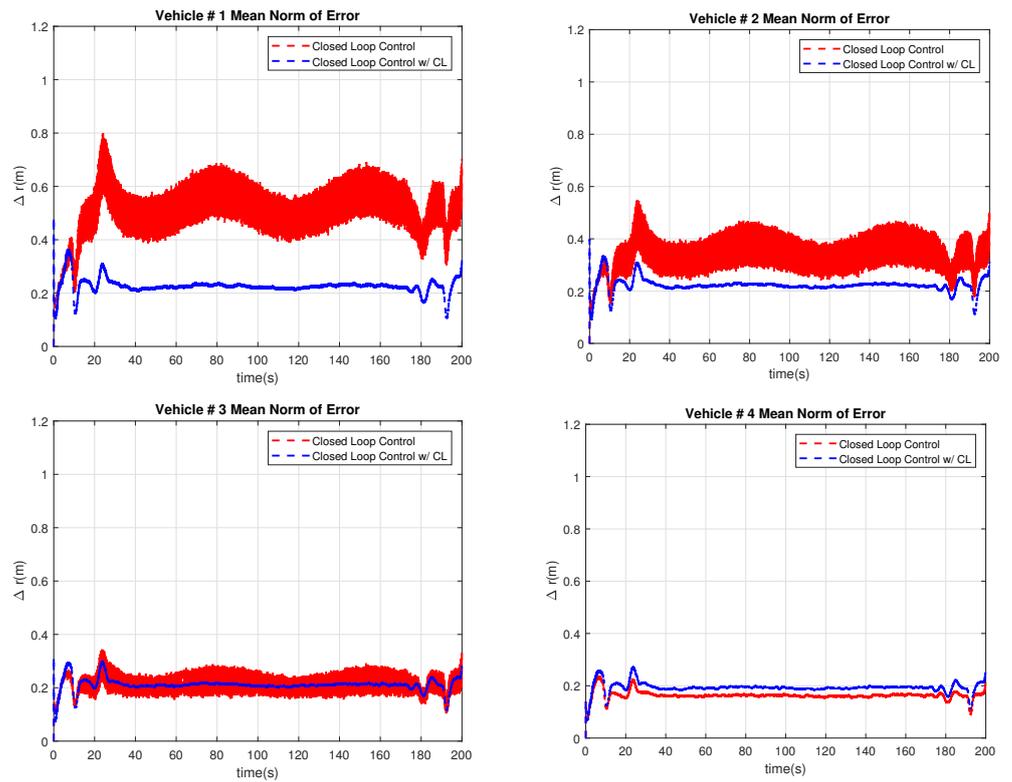


Figure 12. Comparison of the position error norms of Vehicles 1–4 using closed–loop control with and without CL for Case 1 with continuously available absolute position measurements for Vehicle 4.

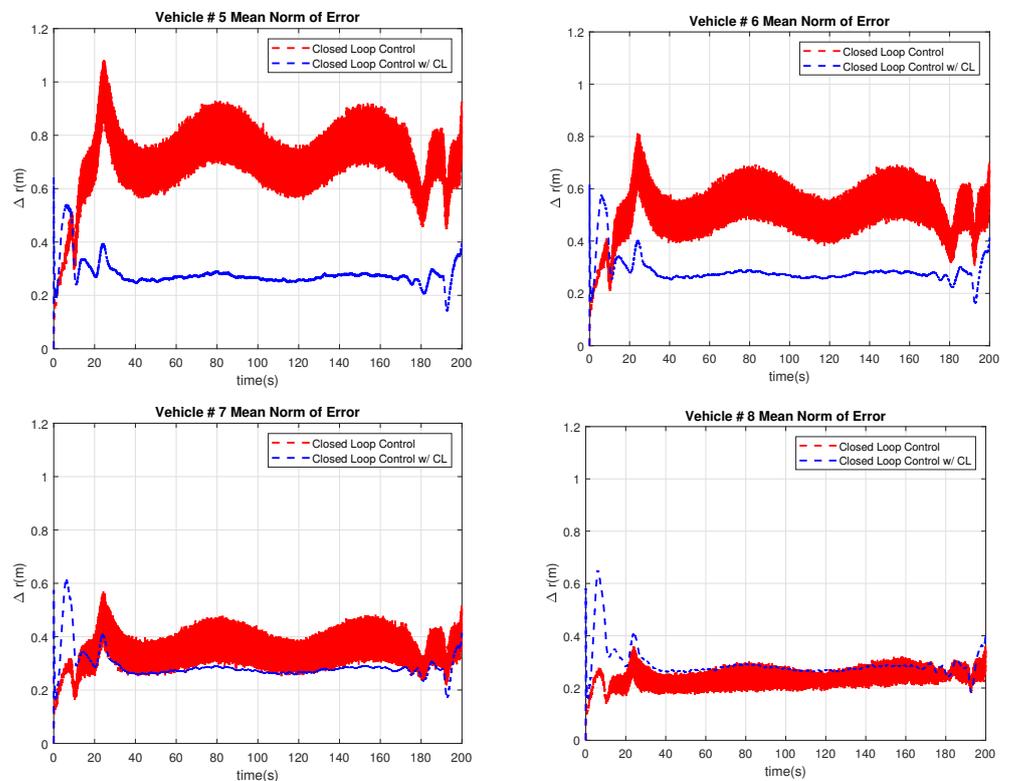


Figure 13. Comparison of the position error norms of Vehicles 5–8 using closed–loop control with and without CL for Case 1 with continuously available absolute position measurements for Vehicle 4.

Case 2 is similar to Case 1 except that Vehicle 4 has intermittent access to its own absolute position measurements. We ensured that the vehicle alternates between 40 s of access to its absolute position measurements followed by lack of access for another 40 s. Figures 14 and 15 present the time history of the position error norms for Vehicles 1–4 and 5–8, respectively, for Case 2. The purple bands represent the “GPS” access afforded to Vehicle 4, and the blank spaces in between represent the “GPS” outages. In this case, we can make similar conclusions as those of Case 1: Vehicle 4 having access to its own absolute measurements does not benefit from CL, the noise in state estimation is significantly reduced for all other vehicles, and the benefit of using CL becomes more pronounced as we go down the network from the lead vehicle with absolute measurement. In addition, any time Vehicle 4 has access to its own absolute position, there is significant reduction in position errors whether CL is employed or not.

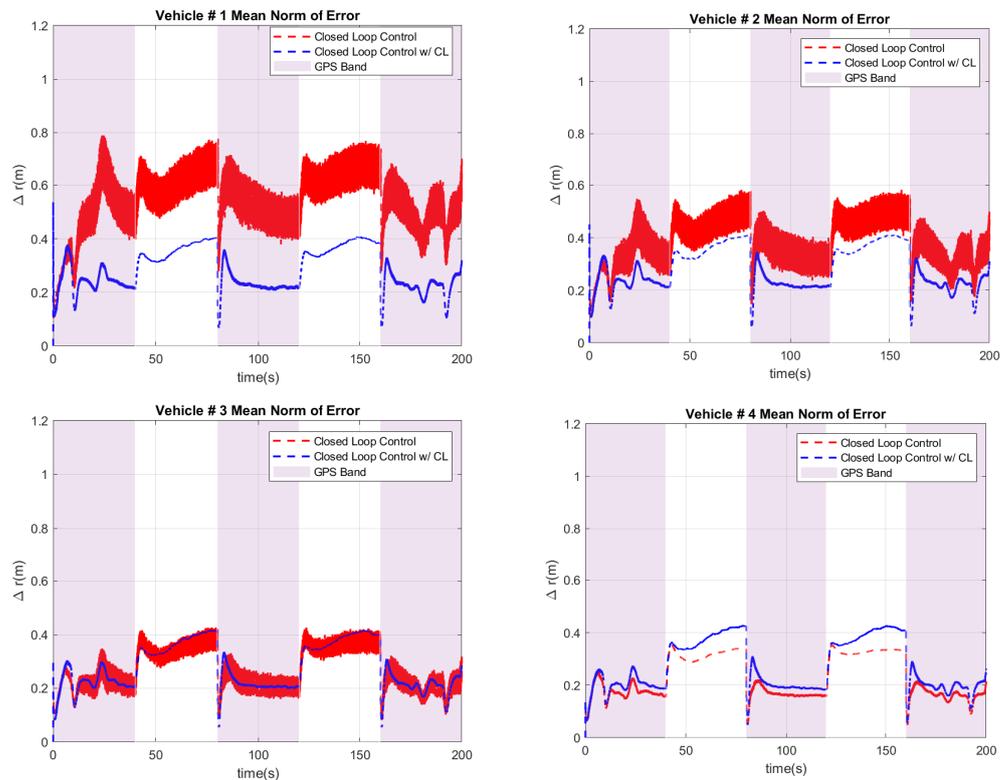


Figure 14. Comparison of the position error norms of Vehicles 1–4 using closed-loop control with and without CL for Case 2 with intermittently available absolute position measurements for Vehicle 4.

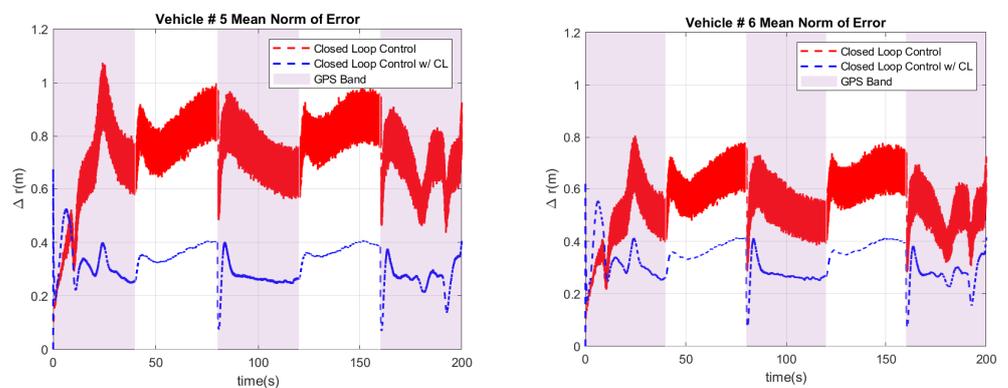


Figure 15. Cont.

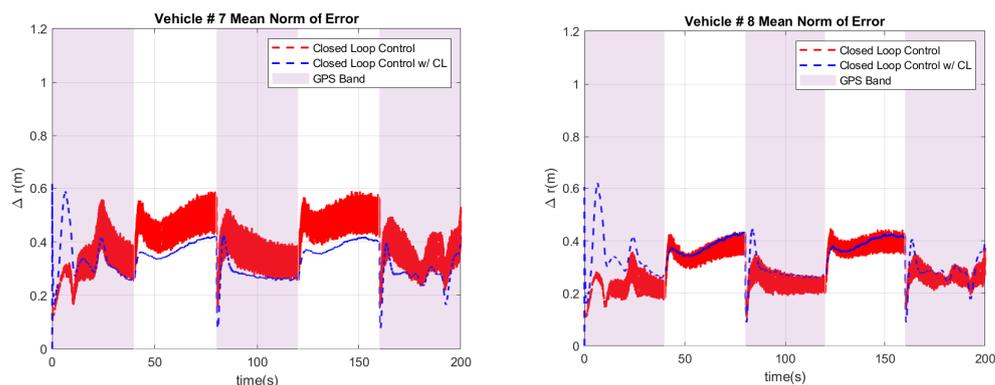


Figure 15. Comparison of the position error norms of Vehicles 5–8 using closed–loop control with and without CL for Case 2 with intermittently available absolute position measurements for Vehicle 4.

7. Conclusions

We presented the nonlinear state and relative measurement equations and their linearized forms for networks of 3D and 2D vehicles and implemented a centralized EKF-based CL algorithm for such networks. An exponentially stable closed-loop nonlinear control was presented that required state estimation for feedback. We dove into the synergistic relationship between CL and closed-loop control within vehicle networks and explored the impact of cooperative localization with consistent and intermittent GPS access. After considering both small and large vehicle networks, we have concluded that cooperative localization can significantly improve the performance of closed-loop control in terms of both noise and error reduction for all vehicles that do not have access to their own absolute position measurement, with the former reducing control effort and the latter helping with position accuracy. Furthermore, the reduction in error becomes more significant as we go down the communication link. Finally, we concluded that cooperative localization is not useful for vehicles that have direct access to their own absolute position measurements through, for example, GPS. Hence, we suggest not updating the position of such vehicles with cooperative localization when implementing the algorithm. These findings have further illuminated the intricate dynamics of cooperative localization and closed-loop control, contributing to a comprehensive understanding of their practical implications and the optimal strategies for deployment in diverse vehicle networks.

Author Contributions: All authors contributed to the study conception and design. Material preparation, data collection, and analysis were performed by J.C.O. The first draft of the manuscript was written by J.C.O. and H.A. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: The datasets generated during and/or analyzed during the current study are available from the corresponding author on reasonable request.

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

Abbreviations

The following abbreviations are used in this manuscript:

CL	Cooperative Localization
2D	Two-Dimensional
3D	Three-Dimensional
EKF	Extended Kalman Filter
UKF	Unscented Kalman Filter
GPS	Global Positioning System
SLAM	Simultaneous Localization and Mapping

References

1. Kia, S.S.; Rounds, S.; Martinez, S. Cooperative Localization for Mobile Agents: A Recursive Decentralized Algorithm Based on Kalman-Filter Decoupling. *IEEE Control Syst. Mag.* **2016**, *36*, 86–101. [[CrossRef](#)]
2. Gamini Dissanayake, M.W.M.; Newman, P.; Clark, S.; Durrant-Whyte, H.F.; Csorba, M. A solution to the simultaneous localization and map building (SLAM) problem. *IEEE Trans. Robot. Autom.* **2001**, *17*, 229–241. [[CrossRef](#)]
3. Bibuli, M.; Bruzzone, G.; Caccia, M.; Gasparri, A.; Priolo, A.; Zereik, E. Swarm-based path-following for cooperative unmanned surface vehicles. *Proc. Inst. Mech. Eng. Part M J. Eng. Marit. Environ.* **2014**, *228*, 192–207. [[CrossRef](#)]
4. Liu, K.; Yang, P.; Jiao, L.; Wang, R.; Yuan, Z.; Dong, S. Antisaturation fixed-time attitude tracking control based low-computation learning for uncertain quadrotor UAVs with external disturbances. *Aerosp. Sci. Technol.* **2023**, *142*, 108668. [[CrossRef](#)]
5. Roumeliotis, S.I.; Bekey, G.A. Distributed multirobot localization. *IEEE Trans. Robot. Autom.* **2002**, *18*, 781–795. [[CrossRef](#)]
6. Mayle, M.N.; Sharma, R. Cooperative Localization in a GPS-Limited Environment Using Inter-Vehicle Range Measurements for a System of Multiple, Non-Homogeneous Vehicles. In Proceedings of the AIAA Scitech 2021 Forum, Virtual Event, 11–15, 19–21 January 2021; American Institute of Aeronautics and Astronautics Inc.: Reston, VA, USA, 2021; pp. 1–18. [[CrossRef](#)]
7. Panzner, S.; Pascucci, F.; Sciacicco, L.; Setola, R. Distributed multi-robot localization. *Robot. Concepts Methodol. Tools Appl.* **2013**, *1*, 391–406. [[CrossRef](#)]
8. Carrillo-Arce, L.C.; Nerurkar, E.D.; Gordillo, J.L.; Roumeliotis, S.I. Decentralized multi-robot cooperative localization using covariance intersection. In Proceedings of the 2013 IEEE/RSJ International Conference on Intelligent Robots and Systems, Tokyo, Japan, 3–7 November 2013; pp. 1412–1417. [[CrossRef](#)]
9. Wanasinghe, T.R.; Mann, G.K.I.; Gosine, R.G. Decentralized Cooperative Localization Approach for Autonomous Multirobot Systems. *J. Robot.* **2016**, *2016*, 2560573. [[CrossRef](#)]
10. Nerurkar, E.D.; Roumeliotis, S.I. Asynchronous multi-centralized cooperative localization. In Proceedings of the IEEE/RSJ 2010 International Conference on Intelligent Robots and Systems, IROS 2010—Conference Proceedings, Taipei, Taiwan, 18–22 October 2010; pp. 4352–4359. [[CrossRef](#)]
11. Zhu, J.; Kia, S.S. Decentralized Cooperative Localization with LoS and NLoS UWB Inter-Agent Ranging. *IEEE Sens. J.* **2022**, *22*, 5447–5456. [[CrossRef](#)]
12. Zhu, P.; Geneva, P.; Ren, W.; Huang, G. Distributed Visual-Inertial Cooperative Localization. In Proceedings of the IEEE International Conference on Intelligent Robots and Systems, Prague, Czech Republic, 27 September–1 October 2021; Institute of Electrical and Electronics Engineers Inc.: Piscataway, NJ, USA, 2021; pp. 8714–8721. [[CrossRef](#)]
13. Wang, D.; Qi, H.; Lian, B.; Liu, Y.; Song, H. Resilient Decentralized Cooperative Localization for Multisource Multirobot System. *IEEE Trans. Instrum. Meas.* **2023**, *72*, 1–13. [[CrossRef](#)]
14. Liu, Y.; Wang, Y.; Shen, Y.; Shi, X. Hybrid TOA-AOA WLS Estimator for Aircraft Network Decentralized Cooperative Localization. *IEEE Trans. Veh. Technol.* **2023**, *72*, 9670–9675. [[CrossRef](#)]
15. Kia, S.S.; Hechtbauer, J.; Gogokhiya, D.; Martinez, S. Server-Assisted Distributed Cooperative Localization over Unreliable Communication Links. *IEEE Trans. Robot.* **2018**, *34*, 1392–1399. [[CrossRef](#)]
16. Wang, L.; Zhang, T.; Gao, F. Distributed cooperative localization with lower communication path requirements. *Robot. Auton. Syst.* **2015**, *79*, 26–39. [[CrossRef](#)]
17. Wang, X.; Guo, Y.; Cao, J.; Wu, M.; Sun, Z.; Lv, C. Simplify Belief Propagation and Variation Expectation Maximization for Distributed Cooperative Localization. *Appl. Sci.* **2022**, *12*, 3851. [[CrossRef](#)]
18. Huang, Y.; Xue, C.; Zhu, F.; Wang, W.; Zhang, Y.; Chambers, J.A. Adaptive Recursive Decentralized Cooperative Localization for Multirobot Systems with Time-Varying Measurement Accuracy. *IEEE Trans. Instrum. Meas.* **2021**, *70*, 1–25. [[CrossRef](#)]
19. Xin, J.; Xie, G.; Yan, B.; Shan, M.; Li, P.; Gao, K. Multimobile Robot Cooperative Localization Using Ultrawideband Sensor and GPU Acceleration. *IEEE Trans. Autom. Sci. Eng.* **2022**, *19*, 2699–2710. [[CrossRef](#)]
20. Lin, J.; Lou, Z.; Zweigel, R.; Abel, D. Cooperative Localization of a Networked Multi-Vehicle System. *IFAC-PapersOnLine* **2019**, *52*, 67–72. [[CrossRef](#)]
21. Wu, Y.; Peng, B.; Wymeersch, H.; Seco-Granados, G.; Kakkavas, A.; Garcia, M.H.C.; Stirling-Gallacher, R.A.; Casta, M.H.; Stirling-Gallacher, R.A. Cooperative Localization with Angular Measurements and Posterior Linearization. In Proceedings of the 2020 IEEE International Conference on Communications Workshops (ICC Workshops), Dublin, Ireland, 7–11 June 2020; pp. 1–6. [[CrossRef](#)]
22. Qu, M.; Lu, H. Cooperative Simultaneous Localization and Mapping with Local Measurements in 2D Space. In Proceedings of the 2021 6th International Conference on Automation, Control and Robotics Engineering (CACRE), Dalian, China, 15–17 July 2021; pp. 509–513. [[CrossRef](#)]
23. Han, Y.; Wei, C.; Li, R.; Wang, J.; Yu, H. A novel cooperative localization method based on IMU and UWB. *Sensors* **2020**, *20*, 467. [[CrossRef](#)]
24. Héry, E.; Xu, P.; Bonnifait, P.; Hery, E.; Xu, P.; Bonnifait, P. Consistent decentralized cooperative localization for autonomous vehicles using LiDAR, GNSS, and HD maps. *J. Field Robot.* **2021**, *38*, 552–571. [[CrossRef](#)]
25. Pires, A.G.; Rezeck, P.A.F.; Chaves, R.A.; Macharet, D.G.; Chaimowicz, L. Cooperative Localization and Mapping with Robotic Swarms. *J. Intell. Robot. Syst. Theory Appl.* **2021**, *102*, 1–23. [[CrossRef](#)]

26. Ben, Y.; Sun, Y.; Li, Q.; Cui, W.; Zhang, Q. A Cooperative Localization Algorithm Considering Unknown Ocean Currents for Multiple AUVs. In *Lecture Notes in Electrical Engineering*; Springer Science and Business Media Deutschland GmbH: Singapore, 2022; Volume 861, pp. 1379–1387. [[CrossRef](#)]
27. Shan, X.; Cabani, A.; Chafouk, H. Cooperative Localization Based on GPS Correction and EKF in Urban Environment. In Proceedings of the 2022 2nd International Conference on Innovative Research in Applied Science, Engineering and Technology, IRASET 2022, Meknes, Morocco, 3–4 March 2022; Institute of Electrical and Electronics Engineers Inc.: Piscataway, NJ, USA, 2022. [[CrossRef](#)]
28. Oliveros, J.C.; Ashrafiuon, H. Application and Assessment of Cooperative Localization in Three-Dimensional Vehicle Networks. *Appl. Sci.* **2022**, *12*, 11805. [[CrossRef](#)]
29. Oliveros, J.C.; Ashrafiuon, H. Cooperative Localization Using the 3D Euler–Lagrange Vehicle Model. *Guid. Navig. Control* **2022**, *2*, 2250018. [[CrossRef](#)]
30. Miller, A.; Rim, K.; Chopra, P.; Kelkar, P.; Likhachev, M. Cooperative Perception and Localization for Cooperative Driving. In Proceedings of the IEEE International Conference on Robotics and Automation, Paris, France, 31 May–31 August 2020; Institute of Electrical and Electronics Engineers Inc.: Piscataway, NJ, USA, 2020; pp. 1256–1262. [[CrossRef](#)]
31. Ilaya, O. Cooperative Control for Multi-Vehicle Swarms. Ph.D. Thesis, RMIT University, Melbourne, Australia, 2009.
32. Yin, T.; Zou, D.; Lu, X.; Bi, C. A Multisensor Fusion-Based Cooperative Localization Scheme in Vehicle Networks. *Electronics* **2022**, *11*, 603. [[CrossRef](#)]
33. Ren, W.; Beard, R.W. Distributed consensus in multi-vehicle cooperative control: Theory and applications. In *Communications and Control Engineering*; Number 9781848000148 in Communications and Control Engineering; Springer: London, UK, 2008; pp. 1–315. [[CrossRef](#)]
34. Betsch, P.; Siebert, R. Rigid body dynamics in terms of quaternions: Hamiltonian formulation and conserving numerical integration. *Int. J. Numer. Methods Eng.* **2009**, *79*, 444–473. [[CrossRef](#)]
35. Ashrafiuon, H.; Nersesov, S.; Clayton, G. Trajectory Tracking Control of Planar Underactuated Vehicles. *IEEE Trans. Autom. Control* **2017**, *62*, 1959–1965. [[CrossRef](#)]
36. Tang, P.; Lin, D.; Zheng, D.; Fan, S.; Ye, J. Observer based finite-time fault tolerant quadrotor attitude control with actuator faults. *Aerosp. Sci. Technol.* **2020**, *104*, 105968. [[CrossRef](#)]
37. Poultney, A.; Kennedy, C.; Clayton, G.; Ashrafiuon, H. Robust Tracking Control of Quadrotors Based on Differential Flatness: Simulations and Experiments. *IEEE/ASME Trans. Mechatron.* **2018**, *23*, 1126–1137. [[CrossRef](#)]

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