

ORIGINAL ARTICLES

Set-theoretic Localization for Mobile Robots with Infrastructure-based Sensing

Xiao Li | Yutong Li | Nan Li | Anouck Girard | Ilya Kolmanovsky

¹Department of Aerospace Engineering, The University of Michigan - Ann Arbor, Michigan, USA

Correspondence

Xiao Li, Aerospace, Robotics, and Controls Laboratory, the University of Michigan - Ann Arbor, MI-48109, USA. Email: hsiaoli@umich.edu

Funding Information

This research was supported by the National Science Foundation under Award ECCS-1931738.

Summary

In this paper, we propose a set-membership based localization approach for mobile robots using infrastructure-based sensing. Under an assumption of known uncertainties bounds of the noise in the sensor measurement and robot motion models, the proposed method computes uncertainty sets that over-bound the robot 2D body and orientation via set-valued motion propagation and subsequent measurement update from infrastructure-based sensing. We establish theoretical properties and computational approaches for this set-theoretic localization method and illustrate its application to an automated valet parking example in simulations, and to omnidirectional robot localization problems in real-world experiments. With deteriorating uncertainties in system parameters and initialization parameters, we conduct sensitivity analysis and demonstrate that the proposed method, in comparison to the FastSLAM, has a milder performance degradation, thus is more robust against the changes in the parameters. Meanwhile, the proposed method can provide estimates with smaller standard deviation values.

KEYWORDS:

localization, set-membership, mobile robot

1 | INTRODUCTION

One of the major challenges for navigating mobile robots safely is in their accurate and reliable localization¹. A promising approach is to leverage the infrastructure-based sensing and wireless communications/V2X². With the increasing computational capability of hardware, real-time simultaneous localization and mapping (SLAM) has been more widely adopted for mobile robot localization tasks in unmapped environments³. In particular, with prior knowledge of the surroundings, the infrastructure-based SLAM is an appealing centralized localization approach as it reduces computational burden by treating individual agent's localization tasks independently⁴. However, quantification of the localization uncertainties is generally handled by the estimation of confidence intervals or ellipsoids within probability-based methods, e.g. Bayesian filters, particle filters¹, etc. Adequate explicit uncertainty bounds, which are crucial for the operation of safety-critical systems, as illustrated by the following mobile robot localization example in Fig. 1, could be difficult to generate via probabilistic methods.

Specifically, suppose a centralized closed-circuit television (CCTV) system that collects measurements associated with the robot is set up as shown in Fig. 1, and suppose the FastSLAM⁵ based on particle filtering is used to estimate the area covered by the robot's body. As shown in Fig. 1, localization results with fewer particles tend to underestimate the area. If we further use the estimated area in a planning module⁶, the one which fails to contain the entire robot body may eventually cause a collision. Though the estimated areas with larger numbers of particles, e.g. 500 and 1000, over-bound the robot body, the increased

This is the author manuscript accepted for publication and has undergone full peer review but has not been through the copyediting, typesetting, pagination and proofreading process, which may lead to differences between this version and the Version of Record. Please cite this article as doi: 10.1002/adc2.117

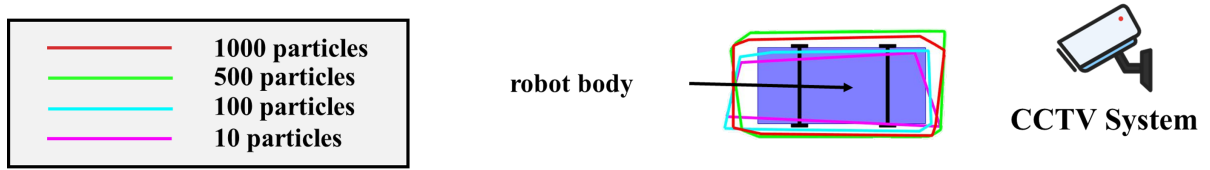


FIGURE 1 FastSLAM estimated robot body area.

sampling and computation burden impedes the online deployment of the FastSLAM. In fact, the estimated area is guaranteed to contain the robot body if and only if we sample an infinite number of particles. We also note that statistical properties of the noise and uncertainty on which probabilistic estimates depend are often assumed; however, they may not necessarily hold in practice.

In this paper, in order to obtain quantitative deterministic uncertainty bounds on estimated states unavailable with probabilistic approaches, we extend the set-theoretic localization approach⁷ to the infrastructure based sensing setting^{8,9,10}. The proposed method can be readily adapted to both camera and lidar sensor systems. With the assumption of bounded robot dynamics uncertainty and measurement noise, the proposed approach guarantees that the actual states are necessarily within the estimated uncertainty sets, which provides a desired quantitative uncertainty bound on the estimated states. Polytopes are used to approximate the uncertainty sets to reduce conservativeness. We use the automated valet parking as an example to validate the effectiveness of the proposed method in simulations, and compare the results with the ones using the FastSLAM. Moreover, we also demonstrate that the proposed algorithm can be readily applied to real-world systems via indoor hardware experiments on a mobile robot.

The main contributions of this paper are as follows: (1) We extend the existing set-theoretic localization approach⁷ to an infrastructure-based sensing setting. (2) We use polytopes to approximate the uncertainty sets which reduces conservativeness as compared to boxes⁷ and is still computationally efficient due to low dimensional characteristics of the problem. (3) We demonstrate the proposed algorithm in a simulated auto-valet parking and a real-world omnidirectional robot localization applications. (4) In a sensitivity analysis, compared to the FastSLAM, we demonstrate the proposed method is more robust and can provide estimates with smaller standard deviation values in presence of changes in the system parameters and initial conditions.

2 | RELATED WORK

Estimation problems¹¹ associated with localization in robotics have been extensively studied. Algorithms such as the classical Kalman filter, particle filter, Bayesian filter, and unscented Kalman filter have been considered for the robotics localization problems^{1,11,12}. A comprehensive review of invariant Kalman filtering that uses the geometric structure of the state space and the dynamics to improve the performance of the extended Kalman filter (EKF) is available in the literature¹³. Our method has a similar structure as the conventional filtering algorithms that predict the states via a dynamics model and subsequently update the prediction using sensor measurements. Unlike the probabilistic methods such as the EKF, the proposed algorithm uses set-valued motion prediction and measurement update to yield a deterministic estimation of uncertainty bounds for robotics localization, mapping, and system state estimation problems^{14,15,16,7,17,18,19}.

In scenarios of real-time exploration tasks, SLAM algorithms are necessary as the environment information is unknown to the robots¹. For SLAM problems, probabilistic methods, for example, EKF SLAM^{20,21} and FastSLAM⁵ have been developed and widely adopted. With the advances in computing power, matrix and graph optimization algorithms²², such as iSAM²³ and GTSAM²⁴, have become feasible for real-time implementation. SLAM algorithms that rely on visual sensors e.g. monocular, stereo, or RGB-D cameras^{25,26,27,28} have also been proposed. In contrast to the classical SLAM problem, our method exploits an infrastructure-based sensing setting, and a prior knowledge of the environment. Moreover, our method is based on centralized localization and hence is able to reduce computational burden by treating individual agents' localization tasks independently⁴.

A wide range of sensors and their combinations have been considered for localization applications. Localization using point clouds generated from lidar sensors has been explored^{29,30} and a more general review of the lidar point cloud registration algorithms was presented by Pomerleau et al.³¹. As a more economical sensor option, visual cameras have been used in visual localization and visual odometry generation³². With the development of machine learning, significant progress in

outdoor visual place recognition and visual localization has been achieved^{33,34,35,36,32}. Meanwhile, sensor fusion such as visual-inertial odometry³⁷ and visual-lidar fusion³⁸ have proved to be effective in mobile robot localization. In our problem setting, as long as the aforementioned sensor systems can provide range and/or angle measurements, they can be directly used as the infrastructure-based sensors in our localization algorithm.

3 | PROBLEM FORMULATION AND PRELIMINARIES

As shown in Fig. 2, we consider a localization system in an $X - Y$ plane $\Omega \in \mathbb{R}^2$ that comprises m ($m \geq 2$) infrastructure-installed sensors, where each individual sensor is located at a certain point in the $X - Y$ plane, and a mobile robot, to which n identical sensor detection markers are attached such that the entire robot body is in the convex hull of the markers. We assume that the sensor system is able to measure the relative angle and/or the relative range of the markers to the sensors. If the sensors' installation positions and orientations were perfectly known, the sensor system could function as a global positioning system for the markers, and subsequently, could localize the robot body area as the convex envelope formed by the markers. However, the actual sensor position and orientation is estimated during the initial calibration³⁹ and these estimates may have errors. Thus, the actual sensor's position and orientation may not be accurately known; in this paper, we assume that the i^{th} sensor's actual position and orientation states are a priori known to belong to an uncertainty set $L_i \in \mathbb{R}^2 \times [-\pi, \pi]$ as shown in Fig. 2.

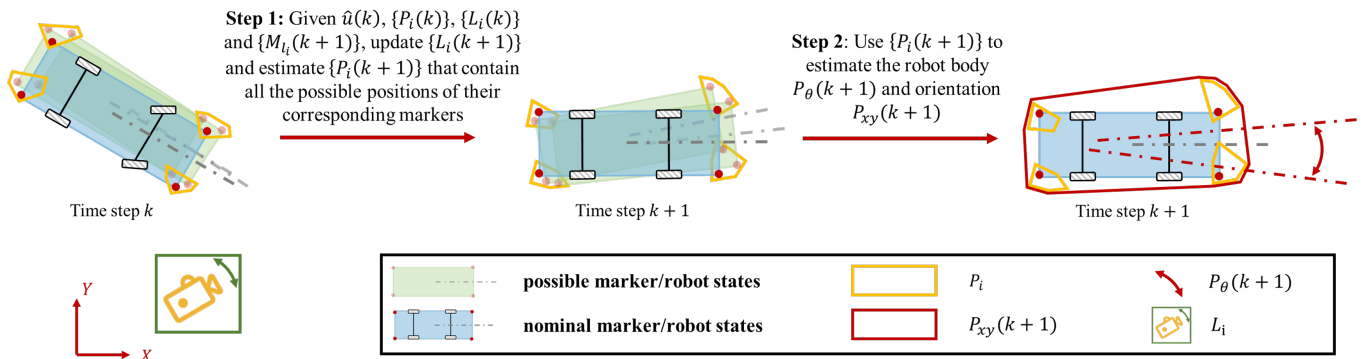


FIGURE 2 Illustrations of the set-theoretic localization approach.

We denote the actual robot body and orientation as $\hat{P}_{xy} \subset \Omega$ and $\hat{p}_\theta \in [-\pi, \pi]$, respectively, and represent the i^{th} marker's actual position by $\hat{p}_i = [\hat{p}_{i,x} \ \hat{p}_{i,y}]^T \in \Omega$. Given robot dynamics, it is possible to derive the equations of motion for the markers. We assume that the marker dynamics can be represented by the following expressions,

$$\hat{p}_i(k+1) = f_i(\hat{p}_i(k), \hat{u}(k) + w_u(k)) + w_f^i(k), \quad i = 1, \dots, n, \quad (1)$$

where the control input $\hat{u}(k) \in \mathbb{R}^{|\mathcal{U}|}$ is subject to an unknown additive noise $w_u(k) \in \mathbb{R}^{|\mathcal{U}|}$, and $w_f^i(k) \in \mathbb{R}^2$ represents the unmodeled disturbance. We assume that the noise $w_u(k)$ and disturbance $w_f^i(k)$ are bounded such that $|w_u(k)| \leq \epsilon^u$, $\|w_f^i(k)\|_\infty \leq \epsilon^f$ with known upper bounds $\epsilon^u \in \mathbb{R}^{|\mathcal{U}|}$ and $\epsilon^f \in \mathbb{R}$. These bounds are characterized from measurements collected during preliminary experimentation with the robots.

We denote the i^{th} sensor's actual state as $\hat{l}_i = [\hat{l}_{i,x} \ \hat{l}_{i,y} \ \hat{l}_{i,\theta}]^T \in L_i$ where the actual installation position is $\hat{l}_{i,xy} = [\hat{l}_{i,x} \ \hat{l}_{i,y}]^T \in \Omega$ and the actual orientation is $\hat{l}_{i,\theta} \in [-\pi, \pi]$. Each single measurement vector $h_{i,j^*}(k) \in \mathbb{R}^{|\mathcal{G}|}$ from the i^{th} sensor obeys the following sensor model,

$$h_{i,j^*}(k) = g_i(\hat{l}_i(k), \hat{p}_{j^*}(k)) + w_g^{i,j^*}(k), \quad (2)$$

where the marker identity $j^* \in \{1, \dots, n\}$ is latent as all markers are identical to the sensor system and $w_g^{i,j^*}(k) \in \mathbb{R}^{|\mathcal{G}|}$ is the unknown additive noise. We assume the noise $w_g^{i,j^*}(k)$ is bounded by a known upper bound $\epsilon^g \in \mathbb{R}^{|\mathcal{G}|}$ such that $|w_g^{i,j^*}(k)| \leq \epsilon^g$. This bound is determined, for instance, from sensor accuracy specification by the sensor manufacturer.

Considering all the aforementioned uncertainties in both the robot dynamics and the sensor measurements, we aim to develop an algorithm that estimates the actual robot body \hat{P}_{xy} and orientation \hat{p}_θ based on the models (1), (2) under the above bounded

noise/uncertainty assumptions. Specifically, at time step k , we treat the actual robot body $\hat{P}_{xy}(k)$, the actual robot orientation $\hat{p}_\theta(k)$, the markers' actual position $\{\hat{p}_i(k)\}_{i=1,\dots,n}$, and the sensors' actual states $\{\hat{l}_i(k)\}_{i=1,\dots,m}$ as unknown and consider the uncertainty sets $\{P_i(k)\}_{i=1,\dots,n}$ and $\{L_i(k)\}_{i=1,\dots,m}$, where $\hat{p}_i(k) \in P_i(k)$ and $\hat{l}_i(k) \in L_i(k)$. Note that sensor uncertainty sets $L_i(0)$ can be generated through the initial calibration³⁹ and the marker uncertainty sets $P_i(0)$ can be initialized by coarse localization using robot on-board sensors⁴⁰. At time step $k+1$, the markers' positions are updated by a given control signal \hat{u}_k through the dynamics in (1). Afterwards, based on (2), each individual i^{th} sensor in the localization system produces a set of measurements $M_i(k+1) = \{h_{i,j^*}(k+1)\}$, $i = 1, \dots, m$.

As shown in Fig 2, the goals of our set-theoretic localization method are as follows:

1. Given control $\hat{u}(k)$, the sensor uncertainty sets $\{L_i(k)\}_{i=1,\dots,m}$, the marker uncertainty sets $\{P_i(k)\}_{i=1,\dots,n}$, and the measurements $\{M_i(k+1)\}_{i=1,\dots,m}$, compute the sensor uncertainty sets $\{L_i(k+1)\}_{i=1,\dots,m}$ and estimate marker uncertainty sets $\{P_i(k+1)\}_{i=1,\dots,n}$ based on (1) and (2) such that $\hat{l}_i \in L_i(k+1)$ for $i = 1, \dots, m$ and $\hat{p}_i(k+1) \in P_i(k+1)$ for $i = 1, \dots, n$.
2. Based on the uncertainty sets $\{L_i(k+1)\}_{i=1,\dots,m}$, $\{P_i(k+1)\}_{i=1,\dots,n}$, estimate two sets $P_{xy}(k+1) \subset \Omega$ and $P_\theta(k+1) \subset [-\pi, \pi]$ such that the robot body is entirely contained in the estimated set, i.e., $\hat{P}_{xy}(k+1) \subset P_{xy}(k+1)$ and the orientation is within the estimated interval, i.e., $\hat{p}_\theta(k+1) \in P_\theta(k+1)$.

4 | MATHEMATICAL MODEL

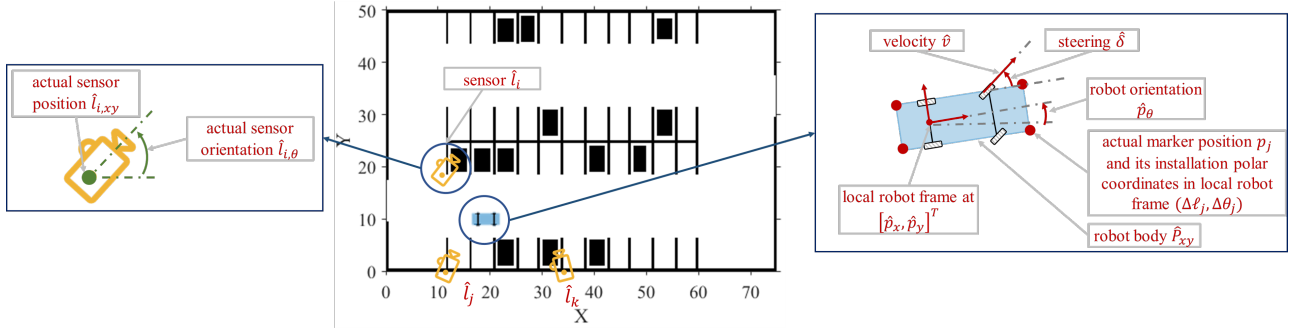


FIGURE 3 Modeling of an infrastructure-based localization system.

In this paper, as shown in Fig. 3, we assume the robot is a front-wheel drive vehicle that is subject to longitudinal velocity and steering control $\hat{u} = [\hat{v} \ \hat{\delta}]^T \in \mathbb{R}^2$, where the unknown additive noise $w_u = [w_v \ w_\delta]^T$ is bounded by $e^u = [e^v \ e^\delta]^T$, i.e., $|w_v| \leq e^v$ and $|w_\delta| \leq e^\delta$. Then, the robot kinematics can be represented by a discrete-time model,

$$\begin{cases} \hat{p}_\theta(k+1) = \hat{p}_\theta(k) + \frac{(\hat{v} + w_v) \cdot dt}{\ell} \cdot \sin(\hat{\delta} + w_\delta) \\ \hat{p}_x(k+1) = \hat{p}_x(k) + (\hat{v} + w_v) \cdot dt \cdot \cos \hat{p}_\theta(k) \cdot \cos(\hat{\delta} + w_\delta) \\ \hat{p}_y(k+1) = \hat{p}_y(k) + (\hat{v} + w_v) \cdot dt \cdot \sin \hat{p}_\theta(k) \cdot \cos(\hat{\delta} + w_\delta) \end{cases} \quad (3)$$

where dt is the sampling period, ℓ is the length of the robot wheelbase, and $[\hat{p}_x \ \hat{p}_y]^T$ is the center of the robot rear wheel axis. In what follows, we derive the markers' equations of motion (1) in Sec. 4.1 and the sensor measurement model (2), which generates angle and range measurements in Sec. 4.2.

4.1 | Robot and Marker Kinematics Models

Derived from (3), and as a realization of (1), the kinematics of the i^{th} marker are represented by

$$\hat{p}_i(k+1) = \hat{p}_i(k) + \begin{bmatrix} d_i(\hat{v} + w_v, \hat{\delta} + w_\delta) \cdot \cos(\theta_i(\hat{v} + w_v, \hat{\delta} + w_\delta, \hat{p}_\theta(k))) \\ d_i(\hat{v} + w_v, \hat{\delta} + w_\delta) \cdot \sin(\theta_i(\hat{v} + w_v, \hat{\delta} + w_\delta, \hat{p}_\theta(k))) \end{bmatrix} + w_f^i(k), \quad (4)$$

where

$$d_i(v, \delta) = v \cdot dt \cdot \sqrt{\left(\frac{\Delta \ell_i \sin \delta}{\ell}\right)^2 + (\cos \delta)^2 - \frac{\Delta \ell_i}{\ell} \cdot \sin \Delta \theta_i \cdot \sin(2\delta)},$$

$$\theta_i(v, \delta, \hat{p}_\theta(k)) = \hat{p}_\theta(k) + \Delta \theta_i + \text{atan2}(\Delta \ell_i \tan \delta - \ell \sin \Delta \theta_i, \ell \cos \Delta \theta_i),$$

$\Delta \ell_i = \sqrt{(\hat{p}_{i,x} - \hat{p}_x)^2 + (\hat{p}_{i,y} - \hat{p}_y)^2}$, $\Delta \theta_i = \text{atan2}(\hat{p}_{i,y} - \hat{p}_y, \hat{p}_{i,x} - \hat{p}_x)$, which are assumed to be given, are the polar coordinates in a local robot frame as shown in Fig. 3. The detailed derivation is available in Appendix A.

4.2 | Sensor Measurement Model

We assume that a sensor, e.g., stereo camera or lidar, is capable of producing angle measurement α and range measurement r . Consequently, each individual measurement is a vector $h_{i,j} = [\alpha_{i,j} \ r_{i,j}]^T$, which contains the angle and range measurements corresponding to an marker of unknown identity $j \in \{1, \dots, n\}$, from the i^{th} sensor. As a realization of (2), the measurement model can be represented by

$$\begin{bmatrix} \alpha_{i,j}(k) \\ r_{i,j}(k) \end{bmatrix} = \begin{bmatrix} \text{atan2}(\hat{p}_{j,y}(k) - \hat{l}_{i,y}(k), \hat{p}_{j,x}(k) - \hat{l}_{i,x}(k)) - \hat{l}_{i,\theta}(k) \\ \sqrt{(\hat{p}_{j,y}(k) - \hat{l}_{i,y}(k))^2 + (\hat{p}_{j,x}(k) - \hat{l}_{i,x}(k))^2} \end{bmatrix} + \begin{bmatrix} w_a^{i,j}(k) \\ w_r^{i,j}(k) \end{bmatrix}, \quad (5)$$

where the unknown additive noise $w_g^{i,j}(k) = [w_a^{i,j}(k) \ w_r^{i,j}(k)]^T \in \mathbb{R}^2$ is bounded by $\epsilon^g = [e^{w_a} \ e^{w_r}]^T \in \mathbb{R}^2$, i.e., $|w_a^{i,j}(k)| \leq \epsilon^{w_a}$ and $|w_r^{i,j}(k)| \leq \epsilon^{w_r}$. Note that in our problem setting, as long as the infrastructure-based sensor can measure the range and/or angle of markers, there is no restriction on the sensor's type, i.e., it can be a monocular camera, lidar, et al. We also note that the marker can be virtual, such as ORB⁴¹ or SIFT⁴² features with object detection⁴³ such as vehicle wheel detection^{44,45} to refine the region of interest.

5 | SET-THEORETIC LOCALIZATION

For simplification of the presentation, in Sec. 5.1 and Sec. 5.2, we assume the sensor system measures only the relative angles from the markers to the sensors, e.g. using monocular cameras, and the measurement is labeled with the corresponding marker identity. We first use the marker kinematics model to propagate the uncertainty sets (Sec. 5.1), then update the sets with corresponding measurements derived from the infrastructure-based sensors (Sec. 5.2). By incorporating the geometrical constraints between individual markers, we can improve the estimation accuracy of the robot body and orientation (Sec. 5.3). Then, we extend our method to the scenario where the actual measurement-to-marker correspondence is latent (Sec. 5.4). An extension of the proposed method to the sensor case with both range and angle measurements, e.g., using stereo cameras, is introduced in Sec. 5.5. We conclude the section with a set over-approximation strategy to simplify the set operations (Sec. 5.6).

5.1 | Motion Propagation

For the sensor uncertainty set propagation, we decompose L_i into two bounded sets $L_{i,xy} \subset \Omega$ and $L_{i,\theta} \subset [-\pi, \pi]$ such that $L_i \subset L_{i,xy} \times L_{i,\theta}$ and \times stands for the Cartesian product, which simplifies the update computations in Sec. 5.2. Then, based on (4), the uncertainty sets P_i of the i^{th} marker and $L_{i,xy}$, $L_{i,\theta}$ of the i^{th} sensor are updated as

$$P_i(k+1|k) = P_i(k) \oplus (D_{i,x}(k) \times D_{i,y}(k)) \oplus \mathcal{B}_2(\epsilon^f), \quad (6)$$

$$\begin{cases} L_{i,xy}(k+1|k) = L_{i,xy}(k) \\ L_{i,\theta}(k+1|k) = L_{i,\theta}(k) \end{cases}, \quad (7)$$

where \oplus denotes the Minkowski sum, $\mathcal{B}_2(\epsilon^f)$ is an ℓ_2 -norm ball of radius ϵ^f , and

$$D_{i,x}(k) = \left\{ dx \in \mathbb{R} : dx = d_i(v, \delta) \cdot \cos(\theta_i(v, \delta, p_\theta)), |v - \hat{v}| \leq \epsilon^v, |\delta - \hat{\delta}| \leq \epsilon^\delta, p_\theta \in P_\theta(k) \right\},$$

$$D_{i,y}(k) = \left\{ dy \in \mathbb{R} : dy = d_i(v, \delta) \cdot \sin(\theta_i(v, \delta, p_\theta)), |v - \hat{v}| \leq \epsilon^v, |\delta - \hat{\delta}| \leq \epsilon^\delta, p_\theta \in P_\theta(k) \right\},$$

are the intervals that bound the displacements of the i^{th} marker along X and Y axis, respectively. We note that $D_{i,x}$, $D_{i,y}$ can be computed using interval arithmetic. Equation (7) reflects the fact that the sensors are stationary.

5.2 | Measurement Update

Given measurements $M_i(k+1) = \{\alpha_{i,j}\}$, $i = 1, \dots, m$, we first update the uncertainty set $L_{i,\theta}$, and then we sequentially update $L_{i,xy}$ and P_j . Based on (5), we update the sensor orientation uncertainty set as follows,

$$L_{i,\theta}(k+1) = L_{i,\theta}(k+1|k) \cap \left(\bigcap_{\alpha_{i,j} \in M_i} [\psi_{i,j}, \phi_{i,j}] \right), \quad (8)$$

where $\psi_{i,j}, \phi_{i,j}$ are derived from each individual measurement $\alpha_{i,j}$ as

$$\begin{aligned} \psi_{i,j} &= \psi(L_{i,xy}, P_j, \alpha_{i,j}) = \beta_{min} - \alpha_{i,j} - \epsilon^{w_a}, \\ \phi_{i,j} &= \phi(L_{i,xy}, P_j, \alpha_{i,j}) = \beta_{max} - \alpha_{i,j} + \epsilon^{w_a}, \end{aligned}$$

and β_{min}, β_{max} are defined as follows

$$\begin{aligned} \beta_{min} &= \min_{\substack{l_{i,xy} \in L_{i,xy}(k+1|k), \\ p_j \in P_j(k+1|k)}} (\text{atan2}(p_{j,y} - l_{i,y}, p_{j,x} - l_{i,x})), \\ \beta_{max} &= \max_{\substack{l_{i,xy} \in L_{i,xy}(k+1|k), \\ p_j \in P_j(k+1|k)}} (\text{atan2}(p_{j,y} - l_{i,y}, p_{j,x} - l_{i,x})) \end{aligned} \quad (9)$$

with bounded sets $L_{i,xy}(k+1|k)$ and $P_j(k+1|k)$. We note that interval $[\psi_{i,j}, \phi_{i,j}]$ is the possible orientation of the i^{th} camera such that the measurement $\alpha_{i,j}$ is plausible, i.e., the measurement noise does not exceed the defined bound ϵ^{w_a} . Please refer to Appendix.B for the detailed proof.

Subsequently, with the updated $L_{i,\theta}(k+1)$ and measurements $M_i(k+1)$, $i = 1, \dots, m$, we estimate the uncertainty sets $L_{i,xy}$ and P_j as follows

$$L_{i,xy}(k+1) = L_{i,xy}(k+1|k) \cap \left(\bigcap_{\alpha_{i,j} \in M_i} (P_j(k+1|k) \oplus L_M(p_j, \alpha_{i,j})) \right), \quad (10)$$

$$P_j(k+1) = P_j(k+1|k) \cap \left(\bigcap_{\substack{i=1, \dots, m, \\ \alpha_{i,j} \in M_i}} ((L_{i,xy}(k+1) \oplus P_M(l_i, \alpha_{i,j}))) \right), \quad (11)$$

where the sets L_M and P_M are defined as

$$L_M(p_j, \alpha_{i,j}) := \{[l_x \ l_y]^T \in \mathbb{R}^2 \mid |\alpha_{i,j} - \text{atan2}(-l_y, -l_x) + \theta_c| \leq \epsilon^{w_a} + \delta\theta_c\}, \quad (12)$$

$$P_M(l_i, \alpha_{i,j}) := \{[p_x \ p_y]^T \in \mathbb{R}^2 \mid |\alpha_{i,j} - \text{atan2}(p_y, p_x) + \theta_c| \leq \epsilon^{w_a} + \delta\theta_c\}, \quad (13)$$

with $\theta_c - \delta\theta_c, \theta_c + \delta\theta_c$ being the minimum and maximum of $L_{i,\theta}(k+1)$, respectively. In fact, in the coordinate frame with p_j as the origin, the set $L_M(p_j, \alpha_{i,j})$ represents a feasible region of l_i for which measurement $\alpha_{i,j}$ is plausible. Analogously, in a reference frame centered at the i^{th} camera, the j^{th} marker should belong to the set $P_M(l_i, \alpha_{i,j})$ given the measurement $\alpha_{i,j}$.

5.3 | Robot Body and Orientation Estimation

Before estimating the robot body and orientation, we can exploit a rigid body constraint for two arbitrary markers $\hat{p}_i, \hat{p}_j \in \{\hat{p}_i\}_{i=1, \dots, n}$ of the form $\|\hat{p}_i - \hat{p}_j\|_2 = r_{ij}$ to further reduce the sizes of the uncertainty sets as follows

$$\begin{aligned} P_i(k+1) &= P_i(k+1) \cap (P_j \oplus \mathcal{B}_2(r_{ij})), \\ P_j(k+1) &= P_j(k+1) \cap (P_i \oplus \mathcal{B}_2(r_{ij})), \end{aligned} \quad (14)$$

where $\mathcal{B}_2(r_{ij})$ represents a ℓ_2 -norm ball of radius r_{ij} .

Based on the set propagation in (6), (7), measurement updates in (19), (20), (21) and set refinement by rigid body constraint in (14), the robot body is over-bounded by a convex envelope $P_{xy}(k+1)$ as shown in Fig. 4 which can be represented as

$$P_{xy}(k+1) := \text{convHull}(\{P_i(k+1)\}_{i=1, \dots, n}), \quad (15)$$

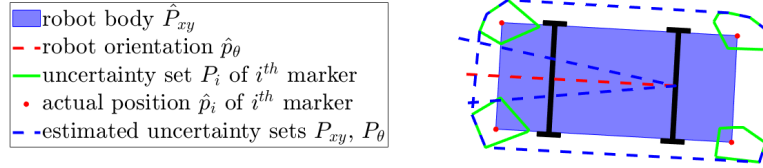


FIGURE 4 Robot body estimation using convex envelope of marker uncertainty sets.

where $\text{convHull}(\cdot)$ is a standard set operation that computes a convex envelope of the given sets. The convex hull computations can be realized using set computational toolboxes, e.g. CORA^{46,47,48}. Consequently, we can update the robot orientation uncertainty set as follows

$$P_\theta = \bigcap_{i,j=1,\dots,n, i \neq j} [\underline{\beta}_{ij} - \Delta\theta_{ij}, \bar{\beta}_{ij} - \Delta\theta_{ij}] \quad (16)$$

where $\Delta\theta_{ij}$ is the offset angle between the vector from \hat{p}_i to \hat{p}_j and the actual robot orientation \hat{p}_θ , and where

$$\underline{\beta}_{ij} = \min_{\substack{l_{i,xy} \in L_{i,xy}(k+1), \\ p_j \in P_j(k+1)}} (\text{atan2}(p_{j,y} - l_{i,y}, p_{j,x} - l_{i,x})),$$

$$\bar{\beta}_{ij} = \max_{\substack{l_{i,xy} \in L_{i,xy}(k+1), \\ p_j \in P_j(k+1)}} (\text{atan2}(p_{j,y} - l_{i,y}, p_{j,x} - l_{i,x})).$$

5.4 | Latent Measurement-to-marker Correspondence

With the assumption of latent measurement-to-marker correspondence, we first deduce the possible correspondence solutions, then, the proposed method presented in Sec. 5.1-5.3 is changed to use modified set updates in (8), (10) and (11). Consider an ordered set of measurements $M_i(k+1) = \{\alpha_{i,j'}^{(q)}(k+1)\}_{q=1,\dots,|M_i|}$ at time step $k+1$ by the i^{th} sensor that contains $|M_i| \leq n$ numbers of measurements, the superscript (q) indicates the order of each individual measurement in the measurement queue and each measurement $\alpha_{i,j'}^{(q)}$ has a unique correspondence to one of the n markers. However, due to the measurement noise in (5) and the state estimation uncertainties in (6) and (7), it's possible that one measurement $\alpha_{i,j'}^{(q)}$ becomes feasible to multiple markers (e.g., $j' \in \{j^*, j_1, j_2, \dots\}$) where the the actual measurement-to-marker correspondence (e.g., $j' = j^*$) is latent. Furthermore, we introduce a matrix C_{l_i} of size $|M_i| \times n$ to include all possible measurement-to-marker correspondence solutions given estimated uncertainty sets where $C_{l_i}(q, j) = 1$ or 0 indicates the q^{th} measurement $\alpha_{i,j'}^{(q)} \in M_i$ can or cannot be a feasible measurement of the j^{th} marker. For instance, suppose we have two measurements for four markers from the i^{th} sensor, and we can represent all possible measurement-to-marker correspondence using a matrix

$$C_{l_i}(k+1) = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 1 & 0 & 1 & 1 \end{bmatrix},$$

which implies the first measurement $\alpha_{i,j_1}^{(1)}$ can be associated with the third marker, i.e., $j_1 = 3$, and the second measurement $\alpha_{i,j_2}^{(2)}$ is a possible measurement corresponding to the first, third and fourth markers, i.e., $j_2 \in \{1, 3, 4\}$. Based on the definition of C_{l_i} , we can deduce possible measurement-to-marker correspondence that are self-consistent and mutually exclusive, i.e., only one entry equals to 1 in each row and column of the matrix C_{l_i} . Again, in the aforementioned example, there are two possible measurement-to-marker correspondences as follows

$$C_{l_i}^{(1)}(k+1) = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix}, \quad C_{l_i}^{(2)}(k+1) = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix},$$

and the actual correspondence must be one of these.

To obtain $C_{l_i}(k+1)$ from $M_i(k+1)$, consider an individual measurement $\alpha_{i,j'}^{(q)} \in M_i(k+1)$, we note that any marker p_j that satisfies the following condition is a candidate to measurement $\alpha_{i,j'}^{(q)}$:

$$P_j(k+1|k) \cap \left(L_{i,xy}(k+1|k) \oplus P'_M(l_i, \alpha_{i,j'}^{(q)}) \right) \neq \emptyset, \quad (17)$$

where

$$P'_M(l_i, \alpha_{i,j'}^{(q)}) := \{[p_x \ p_y]^T \in \mathbb{R}^2 \mid |\alpha_{i,j'}^{(q)} - \text{atan2}(p_y, p_x) + \theta_0| \leq \epsilon^{w_a} + \delta\theta\}, \quad (18)$$

and $\theta_0 - \delta\theta$, $\theta_0 + \delta\theta$ are the minimum and maximum of $L_{i,\theta}(k+1|k)$, respectively. This is due to the fact that the marker associated with the q^{th} measurement is necessarily within $L_{i,xy}(k+1|k) \oplus P'_M(l_i, \alpha_{i,j}^{(q)})$. Then, we can obtain $C_{l_i}(k+1)$, $i = 1, \dots, m$ by examining the criteria in (17) for all measurements obtained from every sensor. We can enumerate through all the possible unique solutions contained in $C_{l_i}(k+1)$ according to the principles of mutual exclusivity and logical self-consistency as in the aforementioned example. This way, for each $M_{l_i}(k+1)$, we are able to generate one or multiple measurement-to-marker correspondence solutions $C_{l_i}^{(\mu)}(k+1)$, $\mu = 1, \dots, c$, which contain the actual measurement-to-marker correspondence.

Given multiple matching solutions $C_{l_i}^{(\mu)}(k+1)$, $\mu \in \{1, \dots, c\}$ for $M_{l_i}(k+1) = \{\alpha_{i,j}^{(q)}(k+1)\}$, the update procedures for $L_{i,\theta}(k+1)$ and $L_{i,xy}(k+1)$ are similar to (8), (10) by applying union operation over all $C_{l_i}^{(\mu)}$, $\mu \in \{1, \dots, c\}$ so that $\hat{l}_{i,\theta}$, $\hat{l}_{i,xy}$ are necessarily within

$$L_{i,\theta}(k+1) = L_{i,\theta}(k+1|k) \cap \left(\bigcup_{\mu=1}^c \left(\bigcap_{\substack{q, \alpha_{i,j}^{(q)} \in M_{l_i}, \\ C_{l_i}^{(\mu)}(q,j)=1}} [\psi(L_{i,xy}, P_j, \alpha_{i,j}^{(q)}), \phi(L_{i,xy}, P_j, \alpha_{i,j}^{(q)})] \right) \right), \quad (19)$$

$$L_{i,xy}(k+1) = L_{i,xy}(k+1|k) \cap \left(\bigcup_{\mu=1}^c \left(\bigcap_{\substack{q, \alpha_{i,j}^{(q)} \in M_{l_i}, \\ C_{l_i}^{(\mu)}(q,j)=1}} (P_j(k+1|k) \oplus L_M(p_j, \alpha_{i,j}^{(q)})) \right) \right), \quad (20)$$

respectively. For update of $P_j(k+1)$, given multiple correspondence solutions $C_{l_i}^{(\mu)}(k+1)$, $\mu = 1, \dots, c$ for M_{l_i} , we suppose there is no corresponding measurement of the j^{th} marker in the actual correspondence $C_{l_i}^{(\mu^*)}$, i.e., $\nexists q \leq n$, such that $C_{l_i}^{(\mu^*)}(q, j) = 1$. In this case, all the measurements in M_{l_i} are irrelevant to the j^{th} marker, which is supposed to be filtered out by the algorithm. In our framework, one can only conclude M_{l_i} certainly contains measurement of the j^{th} marker if there is a corresponding measurement in all correspondence solutions, i.e., $M_{l_i} \in M^j$ and $M^j = \{M_{l_i}, i = 1, \dots, m \mid \forall \mu \in \{1, \dots, c\}, \exists q \leq n, C_{l_i}^{(\mu)}(q, j) = 1\}$. Consider all $M_{l_i} \in M^j$, $i = 1, \dots, m$, the actual marker position \hat{p}_j is necessarily within

$$P_j(k+1) = P_j(k+1|k) \cap \left(\bigcap_{\substack{i=1, \dots, m, \\ M_{l_i} \in M^j}} \left(L_{i,xy}(k+1) \oplus \bigcup_{\substack{\mu=1, \dots, c, \\ \alpha_{i,j}^{(q)} \in M_{l_i}, C_{l_i}^{(\mu)}(q,j)=1}} P_M(l_i, \alpha_{i,j}^{(q)}) \right) \right). \quad (21)$$

5.5 | Sensors with Range and Angle Measurements

With angle and range measurements, the measurement update process of $L_{i,xy}$ and P_j follows Sec. 5.2 and 5.4 where L_M in (12), P_M in (13) and P'_M in (18) are redefined as

$$\begin{aligned} L_M(p_j, \alpha_{i,j}, r_{i,j}) &:= \\ &\left\{ [l_x \ l_y]^T \in \mathbb{R}^2 \mid \begin{array}{l} |\alpha_{i,j} - \text{atan2}(-l_y, -l_x) + \theta_c| \leq e^{w_a} + \delta\theta_c \\ |r_{i,j} - \sqrt{l_x^2 + l_y^2}| \leq e^{w_r} \end{array} \right\}, \\ P'_M(l_i, \alpha_{i,j}, r_{i,j}) &:= \\ &\left\{ [p_x \ p_y]^T \in \mathbb{R}^2 \mid \begin{array}{l} |\alpha_{i,j} - \text{atan2}(p_y, p_x) + \theta_c| \leq e^{w_a} + \delta\theta_c \\ |r_{i,j} - \sqrt{p_x^2 + p_y^2}| \leq e^{w_r} \end{array} \right\}, \\ P_M(l_i, \alpha_{i,j}, r_{i,j}) &:= \\ &\left\{ [p_x \ p_y]^T \in \mathbb{R}^2 \mid \begin{array}{l} |\alpha_{i,j} - \text{atan2}(p_y, p_x) + \theta_0| \leq e^{w_a} + \delta\theta, \\ |r_{i,j} - \sqrt{p_x^2 + p_y^2}| \leq e^{w_r} \end{array} \right\}. \end{aligned}$$

The proposed method in Sec. 5.1-5.5 has the following property (the proof is available in Appendix B):

Proposition 1. Assume $\hat{l}_{i,xy}(0) \in L_{i,xy}(0)$, $\hat{l}_{i,\theta}(0) \in L_{i,\theta}(0)$, $\hat{p}_j(0) \in P_j(0)$ and $i = 1, \dots, m$, $j = 1, \dots, n$. Then, based on the set-theoretic method in (6), (7), (19), (20), (21), (14), (15) and (16), the actual robot body $\hat{P}_{xy}(k)$ and orientation $\hat{p}_\theta(k)$ are confined to the estimated uncertainty sets $P_{xy}(k)$, $P_\theta(k)$ for all $k > 0$, i.e., $\hat{P}_{xy}(k) \subset P_{xy}(k)$, $\hat{p}_\theta(k) \in P_\theta(k)$, $\forall k \geq 0$.

Proposition 1 ensures that the proposed method, based on the aforementioned bounded uncertainty/disturbance assumptions, is able to estimate uncertainty sets that bound the actual robot body and orientation at all times.

5.6 | Uncertainty Set Approximation

In this paper, we approximate the uncertainty sets using polytopes instead of boxes⁷. The benefit of this approximation is that it reduce the conservativeness while this decrease of conservativeness comes at the cost of the increased computational time and effort (which is less of a concern for infrastructure-based computations in this paper as compared to onboard computations). The approximation is illustrated in Fig. 5, where L_M in (12) and P_M in (13) can be represented by two circular sectors in the angular sensor system, e.g., if monocular cameras are used, and two annular sectors in the angular and range sensor system, e.g., if stereo cameras are used. We use convex polygons to over-bound L_M and P_M where $\theta_{1,2} = (\alpha_{i,j} + \theta_c) \pm (\epsilon^{w_a} + \delta\theta_c)$ and $r_{1,2} = r_{i,j} \pm \epsilon^{w_r}$ in Fig. 5 are the angle span and range span of the (circular/annular) sectors, respectively.

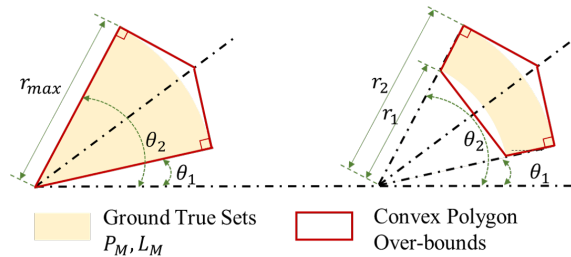


FIGURE 5 Illustrations of set approximations by convex polygons.

6 | SIMULATION AND EXPERIMENTAL RESULTS

In this section, we apply the proposed set-theoretic localization method to an automated valet parking example (Fig. 6). As illustrated in Fig. 6, a simulated parking space equipped with 21 sensors that are assumed to be stereo cameras is built based on the Automated Parking Valet toolbox⁴⁹ in MATLAB. Since the vehicle's equations of motion in the simulation follow the kinematics in (3), we assume no unmodeled disturbance in (4), i.e., $w_f^i = 0$, and other noise terms are sampled randomly within their respectively defined bounds. Meanwhile, each camera has a 70° field of view and a 20 m maximum measurement range. We assume the actual measurement-to-marker correspondence is latent to the sensor system. A vehicle of length $c_l = 4$ m, width $c_w = 1.8$ m and wheelbase $\ell = 2.1$ m is navigating within the parking space tracking a reference trajectory. We assume that four identical markers, denoted as p_i , $i = 1, 2, 3, 4$, are attached to the four vertices of the rectangle that is the vehicle's body. We apply the proposed method to localize the vehicle body \hat{P}_{xy} and orientation \hat{p}_θ , with a sampling period of $dt = 0.5$ s, and the results are compared with the ones using the FastSLAM¹.

The initialization and the detailed operations during the iterations of the proposed algorithm and the FastSLAM are discussed in Sec. 6.1. We quantitatively compare the localization performances of our algorithm against the FastSLAM in Sec. 6.2. Then, the sensitivity analysis results of the proposed algorithm to the system and initialization parameters are discussed in Sec. 6.3. A simulation example, where the proposed method is shown to mitigate the uncertainties in the sensors' orientations and positions, is presented in Sec. 6.4. Moreover, the real-world experimental results with an omnidirectional robot and lidar-based infrastructure sensing system are presented in Sec. 6.5. The code and demonstration videos are available in <https://github.com/XiaoLiSean/SetThmSLAM>.

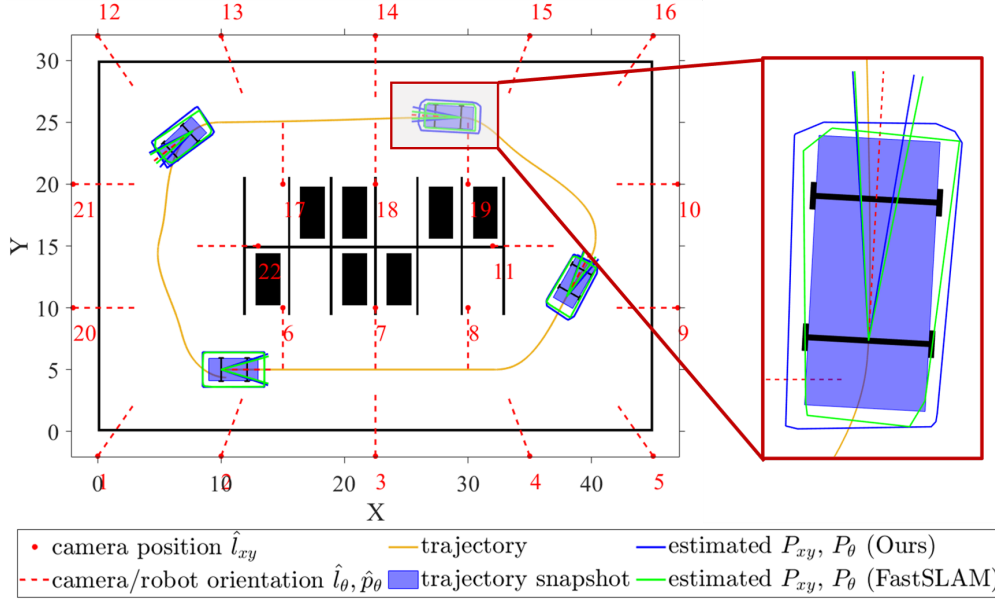


FIGURE 6 Simulation results of the vehicle tracking a reference trajectory in a parking space.

6.1 | Initialization and Iteration of Algorithms

For the initialization of the proposed method, we initialize the uncertainty sets $\{L_{i,xy}\}_{i=1,\dots,m}$, $\{P_i\}_{i=1,\dots,n}$ as boxes, $\{L_{i,\theta}\}_{i=1,\dots,m}$ as intervals centered at their actual states $\{l_{i,xy}\}_{i=1,\dots,m}$, $\{p_i\}_{i=1,\dots,n}$, $\{l_{i,\theta}\}_{i=1,\dots,m}$, respectively. We also note that long term aging or malicious interference with the sensors could lead to the violation of our assumption, i.e., the sensors' initial uncertainty sets $L_{i,xy}(0)$, $L_{i,\theta}(0)$ contain the corresponding actual states, in which case regular maintenance and re-calibration is necessary. The FastSLAM is initialized with 100 particles. Each particle independently stores camera and marker states, i.e., $\{l_{i,xy}^{(s)}, l_{i,\theta}^{(s)}\}_{i=1,\dots,m}$, $\{p_i^{(s)}\}_{i=1,\dots,n}$, $s = 1, \dots, 100$, which are randomly sampled from the aforementioned uncertainty sets, i.e., $\{L_{i,xy}, L_{i,\theta}\}_{i=1,\dots,m}$, $\{P_i\}_{i=1,\dots,n}$.

During iterations, we use CORA^{46,47,48} in MATLAB to implement set operations between polytopes, e.g. Minkowski sums, intersections of polytopes, etc. In the FastSLAM, we estimate P_{xy} using the command `enclosePoints()` in CORA to compute a convex polygon that encloses all marker points $\{p_1^{(s)}, p_2^{(s)}, p_3^{(s)}, p_4^{(s)}\}_{s=1,\dots,100}$ stored in the particles. From the snapshot in Fig. 6, the estimated uncertainty set using the FastSLAM (green solid line) at times fails to contain the entire vehicle body. In contrast, the proposed method guarantees that the vehicle body is always contained within the estimated set (blue solid line), which is consistent with Proposition 1.

6.2 | Estimation Performance

At time step $k = 0$, for the proposed method, we initialize $\{P_i\}_{i=1,\dots,n}$ and $\{L_{i,xy}\}_{i=1,\dots,m}$ as boxes of size $V(P_i) = 1 \text{ m}^2$ and $V(L_{i,xy}) = 0.01 \text{ m}^2$, respectively, and $\{L_{i,\theta}\}_{i=1,\dots,m}$ as intervals of size $V(L_{i,\theta}) = 2 \text{ deg}$. The particles in the FastSLAM randomly sample their states from the initialized boxes and intervals above. The markers' equations of motion are subject to noises w_v, w_δ with bounds $\epsilon^v = 0.1 \text{ m/s}$ and $\epsilon^\delta = 0.5 \text{ deg}$, respectively. We assume the angle and range measurement noises of each cameras are bounded by $\epsilon^{w_a} = 1 \text{ deg}$, $\epsilon^{w_r} = 0.1 \text{ m}$. In addition, two metrics, namely

$$m_1 = \frac{V(P_{xy} \cap \hat{P}_{xy})}{V(P_{xy})},$$

$$m_2 = |\max P_\theta - \hat{p}_\theta| + |\min P_\theta - \hat{p}_\theta|,$$

are used to evaluate the estimation performance of the algorithm. For $m_1 \in [0, 1]$, which is used for evaluating vehicle body estimation performance, the closer the value is to 1, the better is the estimation performance. Similarly for metric $m_2 \in [0, +\infty)$, which is used for evaluating vehicle orientation estimation performance, the closer the value is to 0, the better is the estimation performance.

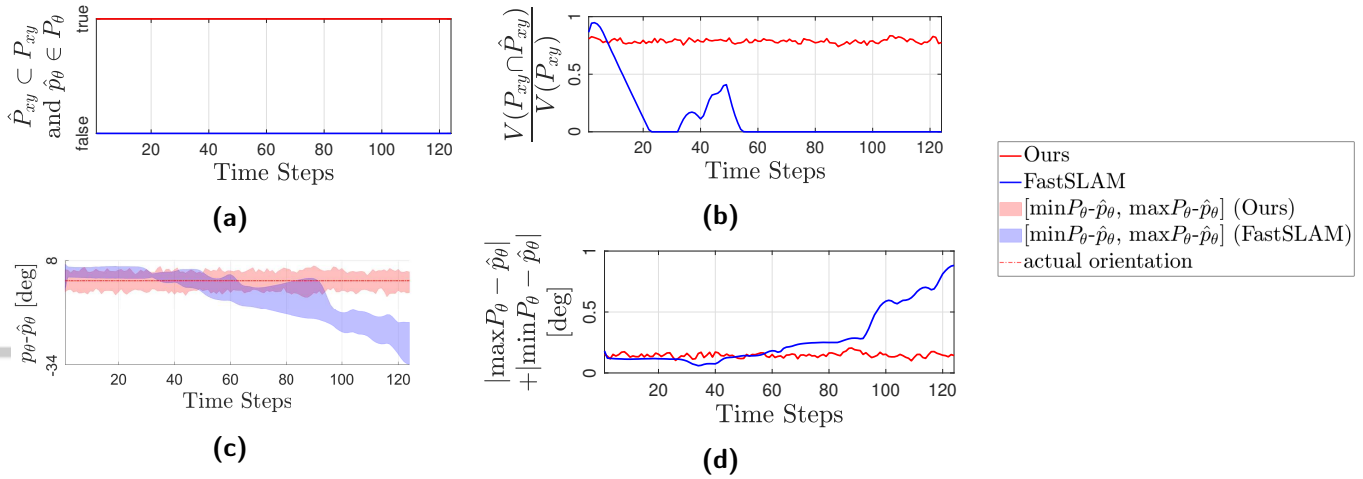


FIGURE 7 Comparison of state estimates between the proposed method and the FastSLAM in a single vehicle path around the parking space: (a) Logical value encoding if the actual vehicle body \hat{P}_{xy} and orientation \hat{p}_θ are in the corresponding estimated sets. (b) Volume of intersection between estimated and actual vehicle body $V(P_{xy} \cap \hat{P}_{xy})$ divided by $V(P_{xy})$. (c) Deviation between the estimated and actual vehicle orientation. The width of the strip represents the size of the estimated uncertainty set. (d) Sum of $|\max P_\theta - \hat{p}_\theta|$ (i.e., the deviation between the maximum in P_θ and the actual vehicle orientation) and $|\min P_\theta - \hat{p}_\theta|$ (i.e., the deviation between the minimum in P_θ and the actual vehicle orientation).

As illustrated in Fig. 7a, our algorithm preserves the claimed property in Proposition 1, i.e., both actual vehicle body and orientation are guaranteed to be contained within the estimated uncertainty sets computed by the proposed method. Furthermore, as shown in Fig. 7b, the proposed method has a more steady and higher value of $\frac{V(P_{xy} \cap \hat{P}_{xy})}{V(P_{xy})}$ compared with the one using the FastSLAM. This is attributed to the containment property of the uncertainty sets estimated by the proposed algorithm, while the uncertainty sets computed by the FastSLAM tend to drift away from the actual ones (in fact, there is a zero overlaps between the estimated and actual vehicle bodies after 60 steps). Similar results are observed in the orientation estimation, as shown in Figs. 7c and 7d, the estimates generated by the proposed algorithm (red strip) contain the red dash line, which indicates $\hat{p}_\theta(k) \in P_\theta(k), \forall k \geq 0$. Meanwhile, the results by the FastSLAM fail to contain it and gradually deviate from the actual orientation line. Differently from the probabilistic methods, e.g., the FastSLAM, the proposed set theoretic localization method computes the uncertainty sets via deterministic set-valued motion propagation and measurement update so that no estimation biases occur, which is easily induced through the weight-based re-sampling procedure in the FastSLAM. This can be further verified by the observation that the strip by the proposed method distributes more evenly around the actual orientation line compared to the one by the FastSLAM in Fig. 7c.

6.3 | Sensitivity Analysis

In this section, we conduct sensitivity analysis of the proposed algorithm to different sensor noise bounds $\epsilon^{w_a}, \epsilon^{w_r}$, initialization uncertainties $V(P_i(0))$ and control signal noise bounds $\epsilon^v, \epsilon^\delta$, and compare the results with the ones using the FastSLAM. Same metrics used in Sec. 6.2 are adopted here for performance comparison between different algorithms. The corresponding results are shown in Figs. 8, 9, 10, respectively, where each data point on the solid line is the mean value of the aforementioned evaluation metrics in Sec. 6.2 and the standard deviation is visualized using shaded strips.

As shown in Figs. 8a and 8c, the proposed algorithm is more robust to the changes in measurement noise bounds in vehicle body estimation, as the mean value of the metric m_1 stays at a steady level while the results from the FastSLAM fluctuate, and the standard deviation of the proposed method is smaller. Similar results can also be observed in vehicle orientation estimation as shown in Figs. 8b and 8d. In fact, the smaller standard deviations and less fluctuated mean values with the changing parameters are due to the fact that our method performs the set estimation in a deterministic way. In contrast, the FastSLAM re-samples particles at each time step, which increases the randomness and leads to larger standard deviation values. We also note that, as shown in Figs. 8b and 8d, the mean values from the proposed method are larger than the ones from the FastSLAM, which implies that the proposed method yields more conservative vehicle orientation estimates to guarantee $\hat{p}_\theta \in P_\theta$.

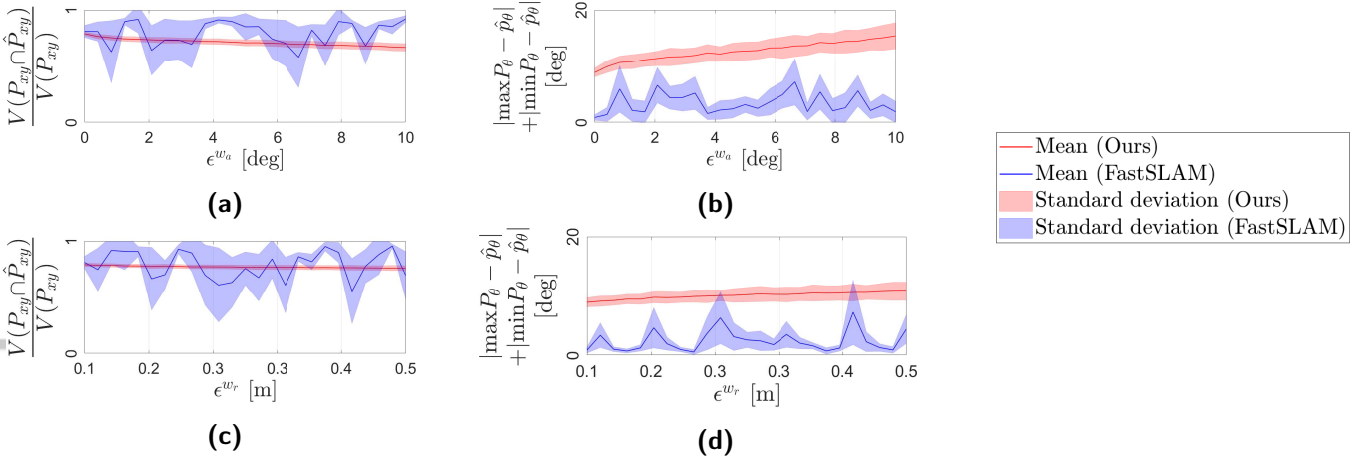


FIGURE 8 Sensitivity analysis to measurement noise bounds: (a) vehicle body estimation with different angle measurement noise bounds ϵ^{w_a} . (b) vehicle orientation estimation with different angle measurement noise bounds ϵ^{w_a} . (c) vehicle body estimation with different range measurement noise bounds ϵ^{w_r} . (d) vehicle orientation estimation with different range measurement noise bounds ϵ^{w_r} .

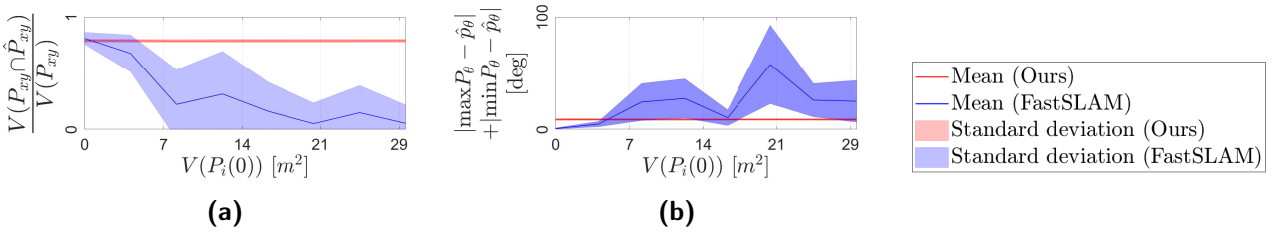


FIGURE 9 Sensitivity analysis to initial marker position uncertainties $V(P_i(0))$: (a) vehicle body estimation with different $V(P_i(0))$. (b) vehicle orientation estimation with different $V(P_i(0))$.

The sensitivity analysis results to varying initial uncertainty set size of marker position are shown in Figs. 9a and 9b. The proposed method yields larger (smaller) mean values and smaller standard deviations in the vehicle body (orientation) estimates, which indicates that the proposed method estimates smaller vehicle body and orientation uncertainty sets compared to the ones by the FastSLAM. The sensitivity analysis results to varying control signal noise are shown in Figs. 10a and 10c. Though the two methods have similar performance in terms of the mean values of m_1 , the proposed method can provide estimates with smaller standard deviation values. Again, the results of vehicle orientation estimation by the proposed method, as shown in Figs. 10b and 10d, are more conservative as a result of enforcing $\hat{p}_\theta \in P_\theta$. In conclusion, against uncertainties in the system parameters and initialization conditions, though the proposed algorithm has more conservative vehicle orientation estimates, it is more robust and ensures that the states are confined to the corresponding uncertainty sets, and it can provide smaller estimation error. For vehicle body estimates, the proposed method has a similar performance as the FastSLAM against measurement and control signal noise bounds, while being more robust to marker initialization uncertainties than the FastSLAM.

6.4 | Accommodating Sensor Uncertainties

At time step $k = 0$, we initialize the proposed method with the same parameters as in Sec. 6.2. Meanwhile, we increase the uncertainties in the initial camera positions and orientations. As shown in Fig. 11, we initialize $\{L_{i,xy}\}_{i=1,\dots,m}$ as boxes of size $V(L_{i,xy}) = 25 \text{ m}^2$, and $\{L_{i,\theta}\}_{i=1,\dots,m}$ as intervals of size $V(L_{i,\theta}) = 20 \text{ deg}$. Fig. 11 visualizes the sensors' and vehicle's uncertainty sets at four different simulation time steps, $k = 0, 5, 10, 15$. The sizes of the camera orientation and position uncertainty sets $L_{i,\theta}$, $L_{i,xy}$, $i = 1, 2, 6, 20$ are decreasing significantly as a result of the update process (19),(20) with small uncertainties in vehicle orientation and body estimation at $k = 0, 5$. However, due to uncertainties in control signals, we observe enlarged vehicle

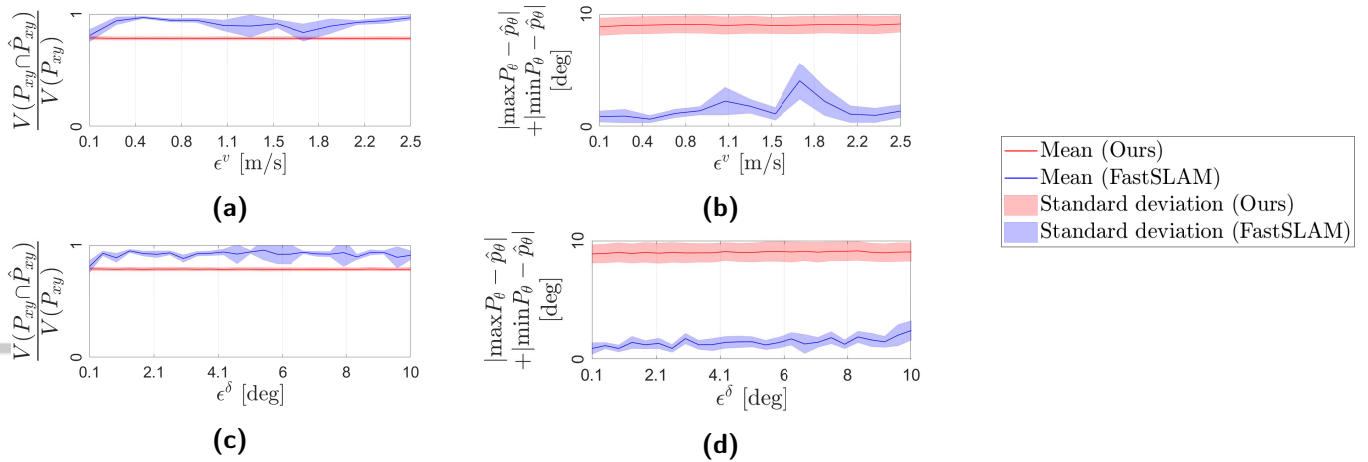


FIGURE 10 Sensitivity analysis to control signal noise bounds: (a) vehicle body estimation with different velocity noise bounds ϵ^v . (b) vehicle orientation estimation with different velocity noise bounds ϵ^v . (c) vehicle body estimation with different steering noise bounds ϵ^δ . (d) vehicle orientation estimation with different steering noise bounds ϵ^δ .

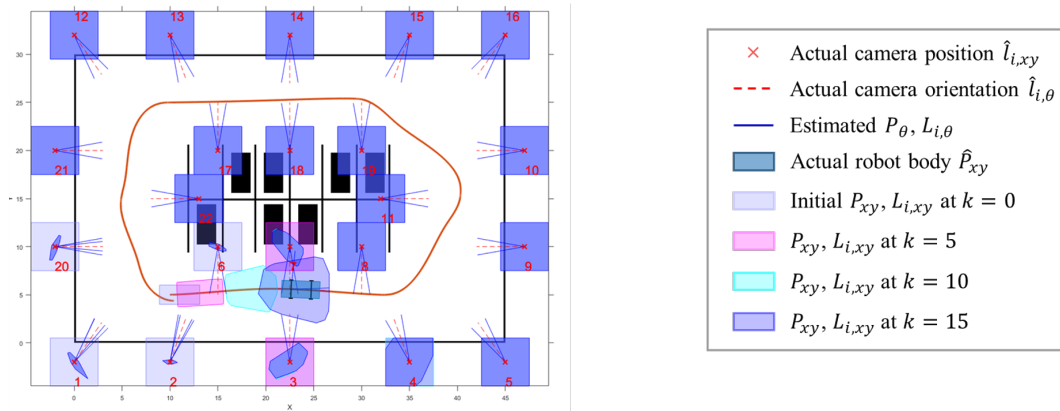


FIGURE 11 Schematic of updating sensor uncertainty sets when the vehicle is accurately localized.

body and orientation uncertainty sets from $k=0$ to $k=15$. Consequently, the updated uncertainty sets $L_{i,\theta}, L_{i,xy}, i=3,4,7$ have larger sizes than those of $L_{i,\theta}, L_{i,xy}, i=1,2,6,20$. Thus, a robot with small orientation and body uncertainty sets is able to mitigate the uncertainties in the sensor orientations and positions. This property can be applied in the sensor calibration process where the robot is well-localized using a third-party global positioning system allowing us to calibrate the sensor parameters and uncertainty sets using the robot localization information.

6.5 | Real-world Experiment Results

To verify the applicability of the proposed method, we set up a sensing system to conduct real-time estimation of an omnidirectional robot body that is visualized as a circle of known radius r in Fig. 12. As shown in Fig. 12, the localization system consists of three lidars (RPLidar A1M8⁵⁰). We attach a vertical bar to the center of the robot and use it as the lidar detection marker \hat{p}_1 such that the task is translated to estimate an uncertainty set P_1 that bounds the robot center, i.e., $\hat{p}_1 \in P_1$. Subsequently, we can estimate the uncertainty set, i.e., $P_{xy} = P_1 \oplus B_2(r)$, that entirely bounds the robot body \hat{P}_{xy} given the circle radius r . Meanwhile,

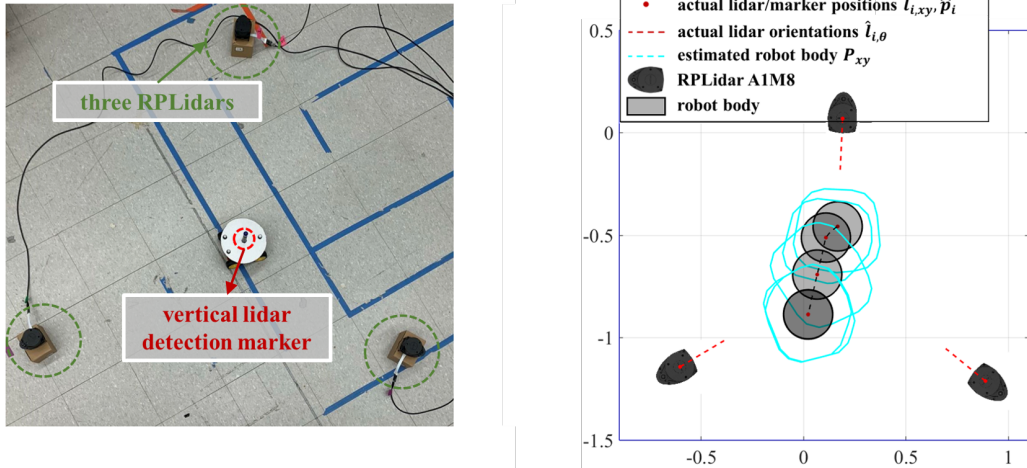


FIGURE 12 Omnidirectional robot localization using lidar measurements: (Left) Photo of the test field with three RPLidars (infrastructure based sensors), and an omnidirectional robot attached with a detection marker. (Right) Test results using the proposed method.

to embed the omnidirectional robot kinematics into the algorithm setting, we define a maximum speed of the robot as v_{max} . Subsequently, in the motion propagation (6), we define $e^f = v_{max} \cdot dt$, set $D_{i,x}(k) \times D_{i,y}(k) = \emptyset$, and overestimate the ball $\mathcal{B}_2(e^f)$ with a square $\mathcal{B}_\infty(e^f)$ in the $X - Y$ plane. The measurement update can then be performed following the proposed method.

We attach three visual detection markers to the robot so that we can obtain the actual position of the robot center from the *OptiTrack* motion capture system as the ground-truth information. The robot is controlled by user via Arduino platform where the maximum speed constraint is enforced. The lidar measurements are transmitted to a master computer, where the proposed localization is performed on MATLAB software, through USB connections and are decoded using 3rd party Python library². The *OptiTrack* measurements is transmitted to the master computer using a Python SDK³ provided by the *OptiTrack*. Based on the aforementioned kinematics assumption of the robot, we note that the synchronization of the control signal and lidar measurements can be conveniently achieved by setting $e^f = v_{max} \cdot dt$ where $dt = 1/f$ and f is the lidar measurement transmission frequency.

We first calibrate the range and angle measurement noise bounds as $\epsilon^{w_r} = 0.073$ m and $\epsilon^{w_a} = 8.05^\circ$, respectively. With a robot trajectory that covers the majority area of the test field, we calibrate the noise bounds as the maximum errors between measurements from *OptiTrack* and the ones from lidars. The robot navigates in the test field with a maximum speed of $v_{max} = 0.10$ m/s. As shown in Fig. 12, the proposed set theoretic localization method guarantees that the estimated uncertainty set (green line) always contains the robot body (circle with a radius of 0.12 m). This result demonstrates the possibility for real-world implementation of the proposed method.⁴

7 | CONCLUSION

In this paper, a set-theoretic localization algorithm that relies on the infrastructure-based sensing has been proposed. The theoretical properties and computational approaches for this set-theoretic localization method have been established. The theoretical properties have also been validated through simulations and real-world experiments. Sensitivity analysis to uncertainties in system parameters and initialization conditions has been conducted. By comparing with the state-of-the-art FastSLAM algorithm, the sensitivity analysis results demonstrated that the proposed method was more robust and ensured that the states were confined to the corresponding uncertainty sets, yet provided smaller estimation errors. Future work will focus on extending the proposed method to localization problems to a higher-dimensional state space, i.e., position and orientation estimation of aerial vehicles.

²The python library is available at <https://pypi.org/project/pyrplidar/>

³The *OptiTrack* SDK is available at <https://optitrack.com/software/netnet-sdk/>

⁴A demonstration video can be found at <https://user-images.githubusercontent.com/58400416/133294083-76bd6d9f-2807-4ab0-ba4e-ffa9abc69788.mp4>



APPENDIX

A MARKER KINEMATICS

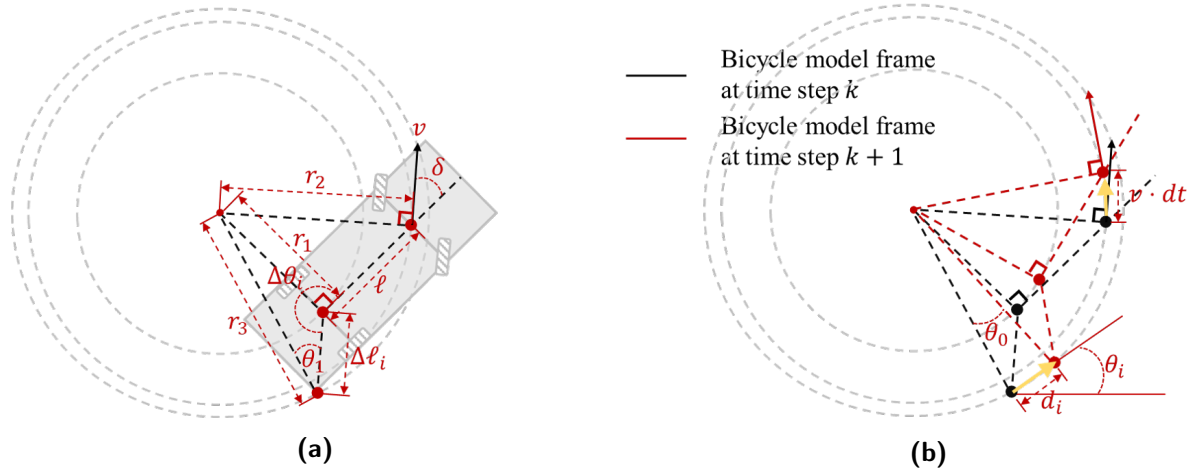


FIGURE A1 Illustration of the i^{th} marker's kinematics.

We use the bicycle model to simplify the kinematics of the vehicle as shown in Fig. A1a. Based on the geometric relationship, we can derive the following equations,

$$r_1 = \frac{\ell}{\tan \delta}, \quad r_2 = \frac{\ell}{\sin \delta},$$

whereby we can obtain the following quantities,

$$\theta_0 = \frac{v \cdot dt}{r_2}, \quad r_3 = \left(r_1^2 + \Delta \ell_i^2 - 2r_1 \Delta \ell_i \cos(\Delta \theta_i - \frac{\pi}{2}) \right)^{1/2}.$$

Then, as shown in Fig. A1b, the displacement of the i^{th} marker is equal to

$$d_i(v, \delta) = \theta_0 r_3 = v \cdot dt \cdot \sqrt{\left(\frac{\Delta \ell_i \sin \delta}{\ell} \right)^2 + (\cos \delta)^2 - \frac{\Delta \ell_i}{\ell} \cdot \sin \Delta \theta_i \cdot \sin(2\delta)}. \quad (\text{A1})$$

Moreover, it can be shown that

$$\sin \theta_1 = \frac{r_1}{\Delta \ell_i} \sin \left(\frac{3\pi}{2} - \Delta \theta_i - \theta_1 \right)$$

which yields

$$\theta_1 = \text{atan2} \left(-\ell \cos \Delta \theta_i, \Delta \ell_i \tan \delta - \ell \sin \Delta \theta_i \right).$$

Afterwards, the angle θ_i of the i^{th} marker satisfies

$$\theta_i(v, \delta, \hat{p}_\theta(k)) = \hat{p}_\theta(k) + \Delta \theta_i + \theta_1 - \frac{3\pi}{2} = \hat{p}_\theta(k) + \Delta \theta_i + \text{atan2} \left(\Delta \ell_i \tan \delta - \ell \sin \Delta \theta_i, \ell \cos \Delta \theta_i \right). \quad (\text{A2})$$

The kinematics model in (4) can be shown from (A1) and (A2).

B PROOF OF PROPOSITION 1

We assume the following condition

$$\begin{aligned} \hat{l}_{i,xy}(k) &\in L_{i,xy}(k), \quad \hat{l}_{i,\theta}(k) \in L_{i,\theta}(k), \quad \hat{p}_j(k) \in P_j(k), \\ i &= 1, \dots, m, \quad j = 1, \dots, n, \end{aligned} \quad (\text{B3})$$

holds for $k = 0$ during initialization. Through the propagation and update of uncertainty sets, if the condition in (B3) being true for $k + 1$ can be induced from same condition being satisfied at time step k , the actual states $\hat{l}_{i,xy}$, $\hat{l}_{i,\theta}$, \hat{p}_j stay in the corresponding uncertainty sets $L_{i,xy}$, $L_{i,\theta}$, P_j , respectively, by principle of induction. Afterward, it's convenient to verify that the robot body and orientation are contained in P_{xy} , P_θ . For simplification, we provide the proof for the monocular camera that the stereo case resembles.

B.1 Motion Propagation

First, we examine the propagation process in (6), (7). The displacement vector in (4) can be shown to satisfy the following property

$$\begin{bmatrix} \hat{d}x \\ \hat{d}y \end{bmatrix} = \begin{bmatrix} \hat{p}_{i,x}(k+1) - \hat{p}_{i,x}(k) \\ \hat{p}_{i,y}(k+1) - \hat{p}_{i,y}(k) \end{bmatrix} \in D_{i,xy}(k)$$

where

$$D_{i,xy}(k) = \left\{ \begin{bmatrix} dx \\ dy \end{bmatrix} \in \mathbb{R}^2 : \begin{bmatrix} dx \\ dy \end{bmatrix} = \begin{bmatrix} d_i(v, \delta) \cdot \cos(\theta_i(v, \delta, p_\theta)) \\ d_i(v, \delta) \cdot \sin(\theta_i(v, \delta, p_\theta)) \end{bmatrix}, |v - \hat{v}| \leq \epsilon^v, |\delta - \hat{\delta}| \leq \epsilon^\delta, p_\theta \in P_\theta(k) \right\}.$$

By the definition of $D_{i,x}(k)$, $D_{i,y}(k)$, we have $D_{i,xy}(k) \subset D_{i,x}(k) \times D_{i,y}(k)$. Considering $w_f^i \in \mathcal{B}_2(\epsilon^f)$ and $\hat{p}_j(k) \in P_j(k)$, the set inclusion $\hat{p}_j(k+1|k) \in P_j(k+1|k)$ is guaranteed by the Minkowski sum in (6). Given that the sensors are stationary, $\hat{l}_{i,xy}(k+1|k) \in L_{i,xy}(k+1|k)$ and $\hat{l}_{i,\theta}(k+1|k) \in L_{i,\theta}(k+1|k)$ follow from (B3) given (7).

B.2 Measurement-to-marker Correspondence

Consider measurements from the i^{th} camera. We now show that the proposed method in Sec. 5.4 can guarantee the possible solutions $C_{l_i}^{(\mu)}$, $\mu = 1, \dots, c$, contain the actual one $C_{l_i}^{(\mu^*)}$. Given $M_{l_i} = \{\alpha_{i,j^*}^{(q)}(k+1)\}$, we suppose the q^{th} measurement $\alpha_{i,j^*}^{(q)}$ is, in fact, a measurement of the j^{*th} marker. In the reference frame taking $\hat{l}_{i,xy}$ as origin, the coordinates of the j^{*th} marker are $\hat{p}_{j^*}(k+1|k) - \hat{l}_{i,xy}(k+1|k)$ and satisfy

$$w_a^{i,j^*}(k+1) = \alpha_{i,j^*}^{(q)}(k+1) - \text{atan2}(\hat{p}_{j^*,y} - \hat{l}_{i,y}, \hat{p}_{j^*,x} - \hat{l}_{i,x}) + \hat{l}_{i,\theta},$$

by (5) where we omit the notations $(k+1|k)$ for simplicity. Since the noise $|w_a^{i,j^*}(k+1)| \leq \epsilon^{w_a}$ is bounded and $\hat{l}_{i,\theta}(k+1|k) \in L_{i,\theta}(k+1|k) \subset [\theta_0 - \delta\theta, \theta_0 + \delta\theta]$, it can be shown from (18) that

$$\hat{p}_{j^*}(k+1|k) - \hat{l}_{i,xy}(k+1|k) \in P'_M(l_i, \alpha_{i,j^*}^{(q)}).$$

Given $\hat{l}_{i,xy}(k+1|k) \in L_{i,xy}(k+1|k)$ from B.1 and equation above, the actual marker position satisfies

$$\begin{aligned} \hat{p}_{j^*}(k+1|k) &= \hat{l}_{i,xy}(k+1|k) + (\hat{p}_{j^*}(k+1|k) - \hat{l}_{i,xy}(k+1|k)) \\ &\in L_{i,xy}(k+1|k) \oplus P'_M(l_i, \alpha_{i,j^*}^{(q)}). \end{aligned}$$

Furthermore, as $\hat{p}_{j^*}(k+1|k) \in P_{j^*}(k+1|k)$ from B.1, the nonempty condition in (17) is satisfied, therefore, $C_{l_i}(q, j^*) = 1$. Same conclusion can be applied to each single measurement in $M_{l_i}(k+1)$. Thus, the actual solution $C_{l_i}^{(\mu^*)}$ is guaranteed to be presented in the correspondence matrix $C_{l_i}(k+1)$. An algorithm examining the logical self-consistency in $C_{l_i}(k+1)$ will keep $C_{l_i}^{(\mu^*)}$ as one of the solutions $C_{l_i}^{(\mu)}$, $\mu \in \{1, \dots, c\}$.

B.3 Measurement Update

In the following discussion, we first present the proof with only one measurement-to-marker solution that is actual, then, the extension to case with multiple correspondences will be presented.

B.3.1 Update of $L_{i,\theta}$

Given only one solution of $M_{l_i}(k+1) = \{\alpha_{i,j}^{(q)}(k+1)\}$ that is actual, we can obtain $\hat{l}_{i,\theta}(k+1) \in [\psi_{i,j}, \phi_{i,j}]$ for each measurement $\alpha_{i,j}^{(q)}(k+1)$ by (5) and (9). Indeed, consider the entire set of measurements, we have

$$\hat{l}_{i,\theta}(k+1) \in \bigcap_{\alpha_{i,j}^{(q)} \in M_{l_i}} [\psi_{i,j}, \phi_{i,j}],$$

which, combining with $\hat{l}_{i,\theta}(k+1) \in L_{i,\theta}(k+1|k)$, proves (8). Moreover, if multiple solutions $C_{l_i}^{(\mu)}$, $\mu = 1, \dots, c$ are given where the actual one $C_{l_i}^{(\mu^*)}$ is contained, the following relationship can also be established

$$\hat{l}_{i,\theta}(k+1) \in \bigcap_{\substack{q, \alpha_{i,j'}^{(q)} \in M_{l_i}, \\ C_{l_i}^{(\mu^*)}(q,j)=1}} [\psi_{i,j}, \phi_{i,j}] \subset \bigcup_{\mu=1}^c \left(\bigcap_{\substack{q, \alpha_{i,j'}^{(q)} \in M_{l_i}, \\ C_{l_i}^{(\mu)}(q,j)=1}} [\psi(L_{i,xy}, P_j, \alpha_{i,j'}^{(q)}), \phi(L_{i,xy}, P_j, \alpha_{i,j'}^{(q)})] \right).$$

Given $\hat{l}_{i,\theta}(k+1) \in L_{i,\theta}(k+1|k)$ and equation above, we can show $\hat{l}_{i,\theta}(k+1) \in L_{i,\theta}(k+1)$ as in (19).

B.3.2 Update of $L_{i,xy}$

Given measurement $\alpha_{i,j}$ and in a reference frame centered at \hat{p}_j , the coordinates of the i^{th} camera $[l'_x \ l'_y]^T = \hat{l}_{i,xy}(k+1|k) - \hat{p}_j(k+1|k)$ satisfy

$$\text{atan2}(-l'_y, -l'_x) = \alpha_{i,j} + \hat{l}_{i,\theta}(k+1|k) - w_a^{i,j}(k+1)$$

by (5). Since the noise $|w_a^{i,j}(k+1)| \leq \epsilon^{w_a}$ and $\hat{l}_{i,\theta}(k+1|k) \in L_{i,\theta}(k+1|k) \subset [\theta_c - \delta\theta_c, \theta_c + \delta\theta_c]$, it can be shown that

$$\hat{l}_{i,xy}(k+1|k) - \hat{p}_j(k+1|k) \in L_M(p_j, \alpha_{i,j})$$

from (12). Given $\hat{p}_j(k+1|k) \in P_j(k+1|k)$ from propagation and equation above, the actual camera position satisfies

$$\begin{aligned} \hat{l}_i(k+1|k) &= \hat{p}_j(k+1|k) + (\hat{l}_{i,xy}(k+1|k) - \hat{p}_j(k+1|k)) \\ &\in P_j(k+1|k) \oplus L_M(p_j, \alpha_{i,j}). \end{aligned}$$

Considering all the measurements, the actual camera position is within the set intersection

$$\bigcap_{\alpha_{i,j} \in M_{l_i}} (P_j(k+1|k) \oplus L_M(p_j, \alpha_{i,j})),$$

which together with $\hat{l}_{i,xy}(k+1) \in L_{i,xy}(k+1|k)$ proves (10).

Similar to B.3.1, if multiple solutions are given including the actual one, the following relationship can also be established

$$\begin{aligned} \hat{l}_{i,xy}(k+1|k) &\in \bigcap_{\substack{q, \alpha_{i,j'}^{(q)} \in M_{l_i}, \\ C_{l_i}^{(\mu^*)}(q,j)=1}} (P_j(k+1|k) \oplus L_M(p_j, \alpha_{i,j'}^{(q)})) \\ &\subset \bigcup_{\mu=1}^c \left(\bigcap_{\substack{q, \alpha_{i,j'}^{(q)} \in M_{l_i}, \\ C_{l_i}^{(\mu)}(q,j)=1}} (P_j(k+1|k) \oplus L_M(p_j, \alpha_{i,j'}^{(q)})) \right). \end{aligned}$$

Eventually, with $\hat{l}_{i,xy}(k+1) \in L_{i,xy}(k+1|k)$, we can prove $\hat{l}_{i,xy}(k+1) \in L_{i,xy}(k+1)$ as in (20).

B.3.3 Update of P_j

It's likely that only a subset of cameras have corresponding measurements of the j^{th} marker. Similarly, we note that

$$\hat{p}_j(k+1) - \hat{l}_{i,xy}(k+1) \in P_M(l_i, \alpha_{i,j}).$$

As $\hat{l}_{i,xy}(k+1) \in L_{i,xy}(k+1)$ from B.3.2, we have

$$\hat{p}_j(k+1) \in L_{i,xy}(k+1) \oplus P_M(l_i, \alpha_{i,j}).$$

Considering M_{l_i} , $i = 1, \dots, m$ that contain measurement of the j^{th} marker and $\hat{p}_j(k+1) \in P_j(k+1|k)$ from B.1, the actual marker position satisfies $\hat{p}_j(k+1) \in P_j(k+1)$ as derived in (11).

Again, given multiple solutions for M_{l_i} , one can only conclude M_{l_i} certainly contains measurements of the j^{th} marker if there is a corresponding measurement in all solutions, i.e., $\forall \mu \in \{1, \dots, c\}, \exists q \leq n, C_{l_i}^{(\mu)}(q, j) = 1$. Given such a set of measurements $M_{l_i} \in M^j$, we notice that

$$\hat{p}_j(k+1) - \hat{l}_{i,xy}(k+1) \in P_M(l_i, \alpha_{i,j}) \subset \bigcup_{\substack{\mu=1, \dots, c, \\ \alpha_{i,j'}^{(q)} \in M_{l_i}, C_{l_i}^{(\mu)}(q, j)=1}} P_M(l_i, \alpha_{i,j'}^{(q)}).$$

Given $\hat{l}_{i,xy}(k+1) \in L_{i,xy}(k+1)$ from B.3.2, we have

$$\hat{p}_j(k+1) \in L_{i,xy}(k+1) \oplus \bigcup_{\substack{\mu=1, \dots, c, \\ \alpha_{i,j'}^{(q)} \in M_{l_i}, C_{l_i}^{(\mu)}(q, j)=1}} P_M(l_i, \alpha_{i,j'}^{(q)}).$$

Consider all $M_{l_i} \in M^j$, $i = 1, \dots, m$ and $\hat{p}_j(k+1) \in P_j(k+1|k)$, the actual marker position $\hat{p}_j(k+1) \in P_j(k+1)$ as derived in (21).

B.4 Robot Body and Orientation Estimation


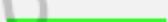
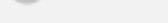
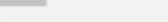
Points that are at most r distance away from p_i locate in $P_i \oplus \mathcal{B}_2(r)$. If $\|\hat{p}_i - \hat{p}_j\|_2 = r_{ij}$, the actual marker position $\hat{p}_j \in P_i \oplus \mathcal{B}_2(r_{ij})$. Thereby, the set refinement by rigid body constrains in (14) preserves the property that $\hat{p}_j \in P_j(k+1)$. Finally, given the assumption of the robot body being in the convex hull of the markers, \hat{P}_{xy} in (15) over-bounds the entire robot body P_{xy} , i.e., $\hat{P}_{xy} \subset P_{xy}$. The proof of the actual robot orientation $\hat{p}_\theta \in P_\theta$ follows a similar procedure as in B.3.1.

References

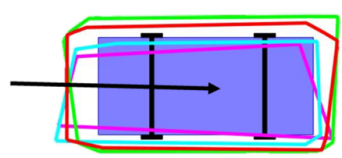
1. Thrun S, Burgard W, Fox D. *Probabilistic robotics*. MIT Press . 2005.
2. Boban M, Kousaridas A, Manolakis K, Eichinger J, Xu W. Connected roads of the future: Use cases, requirements, and design considerations for vehicle-to-everything communications. *IEEE Vehicular Technology Magazine* 2018; 13(3): 110–123.
3. Thrun S. Simultaneous localization and mapping. In: Springer. 2007 (pp. 13–41).
4. Teixeira T, Jung D, Savvides A. Tasking networked cctv cameras and mobile phones to identify and localize multiple people. In: *Proceedings of the 12th ACM International Conference on Ubiquitous Computing*. ; 2010: 213–222.
5. Montemerlo M, Thrun S, Koller D, Wegbreit B, others . FastSLAM: A factored solution to the simultaneous localization and mapping problem. *AAAI Innovative Applications of Artificial Intelligence* 2002; 593598.
6. Ceccarelli N, Di Marco M, Garulli A, Giannitrapani A. A set theoretic approach to path planning for mobile robots. In: . 1. *43rd IEEE Conference on Decision and Control*. ; 2004: 147–152.
7. Di Marco M, Garulli A, Giannitrapani A, Vicino A. A set theoretic approach to dynamic robot localization and mapping. *Autonomous Robots* 2004; 16(1): 23–47.
8. Song B. Cooperative lateral vehicle control for autonomous valet parking. *International Journal of Automotive Technology* 2013; 14(4): 633–640.
9. Sung K, Choi J, Kwak D. Vehicle control system for automatic valet parking with infrastructure sensors. In: *IEEE International Conference on Consumer Electronics (ICCE)*. ; 2011: 567–568.

10. Khalid M, Wang K, Aslam N, Cao Y, Ahmad N, Khan MK. From smart parking towards autonomous valet parking: A survey, challenges and future Works. *Journal of Network and Computer Applications* 2020: 102935.
11. Barfoot TD. *State estimation for robotics*. Cambridge University Press . 2017.
12. Smith A. *Sequential Monte Carlo methods in practice*. Springer Science & Business Media . 2013.
13. Barrau A, Bonnabel S. Invariant kalman filtering. *Annual Review of Control, Robotics, and Autonomous Systems* 2018; 1: 237–257.
14. Hanebeck UD, Schmidt G. Set theoretic localization of fast mobile robots using an angle measurement technique. In: . 2. *Proceedings of IEEE International Conference on Robotics and Automation*. ; 1996: 1387–1394.
15. Alamo T, Bravo JM, Camacho EF. Guaranteed state estimation by zonotopes. *Automatica* 2005; 41(6): 1035–1043.
16. Merhy D, Stoica Maniu C, Alamo T, et al. Guaranteed set-membership state estimation of an octorotor’s position for radar applications. *International Journal of Control* 2020; 93(11): 2760–2770.
17. Di Marco M. *Set-membership estimation techniques for mobile robotics applications*. PhD thesis. Bologna University, ; 2001.
18. Wang H, Kolmanovsky I, Sun J. Zonotope-based recursive estimation of the feasible solution set for linear static systems with additive and multiplicative uncertainties. *Automatica* 2018; 95: 236–245.
19. Kolmanovsky I, Sivergina I, Sun J. Simultaneous input and parameter estimation with input observers and set-membership parameter bounding: Theory and an automotive application. *International Journal of Adaptive Control and Signal Processing* 2006; 20(5): 225–246.
20. Moutarlier P, Chatila R. An experimental system for incremental environment modelling by an autonomous mobile robot. In: Springer. ; 1990: 327–346.
21. Moutarlier P. Stochastic multisensory data fusion for mobile robot location and environment modeling. *Proceedings of International Symposium on Robotics Research* 1989.
22. Grisetti G, Kümmerle R, Stachniss C, Burgard W. A tutorial on graph-based SLAM. *IEEE Intelligent Transportation Systems Magazine* 2010; 2(4): 31–43.
23. Kaess M, Ranganathan A, Dellaert F. iSAM: Incremental smoothing and mapping. *IEEE Transactions on Robotics* 2008; 24(6): 1365–1378.
24. Dellaert F. Factor graphs and GTSAM: A hands-on introduction. tech. rep., Georgia Institute of Technology; : 2012.
25. Mur-Artal R, Montiel JMM, Tardos JD. ORB-SLAM: a versatile and accurate monocular SLAM system. *IEEE Transactions on Robotics* 2015; 31(5): 1147–1163.
26. Engel J, Schöps T, Cremers D. LSD-SLAM: Large-scale direct T monocular SLAM. In: *European Conference on Computer Vision*. ; 2014: 834–849.
27. Davison AJ, Reid ID, Molton ND, Stasse O. MonoSLAM: Real-time single camera SLAM. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 2007; 29(6): 1052–1067.
28. Whelan T, Kaess M, Johannsson H, Fallon M, Leonard JJ, McDonald J. Real-time large-scale dense RGB-D SLAM with volumetric fusion. *The International Journal of Robotics Research* 2015; 34(4-5): 598–626.
29. Wolcott RW, Eustice RM. Fast LIDAR localization using multiresolution Gaussian mixture maps. In: *2015 IEEE international conference on robotics and automation (ICRA)*. ; 2015: 2814–2821.
30. Yoneda K, Tehrani H, Ogawa T, Hukuyama N, Mita S. Lidar scan feature for localization with highly precise 3-D map. In: *2014 IEEE Intelligent Vehicles Symposium Proceedings*. ; 2014: 1345–1350.

31. Pomerleau F, Colas F, Siegwart R. A review of point cloud registration algorithms for mobile robotics. *Foundations and Trends in Robotics* 2015; 4(1): 1–104.
32. Nistér D, Naroditsky O, Bergen J. Visual odometry. In: . 1. *Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition.* ; 2004: I–I.
33. Lowry S, Sünderhauf N, Newman P, et al. Visual Place Recognition: A Survey. *IEEE Transactions on Robotics* 2016; 32(1): 1–19. doi: [10.1109/TRO.2015.2496823](https://doi.org/10.1109/TRO.2015.2496823)
34. Naseer T, Oliveira GL, Brox T, Burgard W. Semantics-aware visual localization under challenging perceptual conditions. In: *IEEE International Conference on Robotics and Automation.* ; 2017: 2614–2620.
35. Pirker K, Rütther M, Bischof H. CD SLAM-continuous localization and mapping in a dynamic world. In: *IEEE/RSJ International Conference on Intelligent Robots and Systems.* ; 2011: 3990–3997.
36. Arandjelović R, Gronat P, Torii A, Pajdla T, Sivic J. NetVLAD: CNN architecture for weakly supervised place recognition. In: *IEEE Conference on Computer Vision and Pattern Recognition.* ; 2016.
37. Leutenegger S, Lynen S, Bosse M, Siegwart R, Furgale P. Keyframe-based visual–inertial odometry using nonlinear optimization. *The International Journal of Robotics Research* 2015; 34(3): 314–334.
38. Debeunne C, Vivet D. A review of visual-LiDAR fusion based simultaneous localization and mapping. *Sensors* 2020; 20(7): 2068.
39. Devarajan D, Cheng Z, Radke RJ. Calibrating distributed camera networks. *Proceedings of the IEEE* 2008; 96(10): 1625–1639.
40. Bostanci B, Tekkok S, Soyunmez E, Oguz-Ekim P, Yeganli F. The lidar and uwb based source localization and initialization algorithms for autonomous robotic systems. In: *2019 11th International Conference on Electrical and Electronics Engineering (ELECO).* ; 2019: 900–904.
41. Rublee E, Rabaud V, Konolige K, Bradski G. ORB: An efficient alternative to SIFT or SURF. In: *IEEE International Conference on Computer Vision.* ; 2011: 2564–2571.
42. Lowe DG. Object recognition from local scale-invariant features. In: . 2. *Proceedings of the 7th IEEE International Conference on Computer Vision.* ; 1999: 1150–1157.
43. Papageorgiou C, Poggio T. A trainable system for object detection. *International Journal of Computer Vision* 2000; 38(1): 15–33.
44. Achler O, Trivedi MM. Camera based vehicle detection, tracking, and wheel baseline estimation approach. In: *Proceedings of the 7th International IEEE Conference on Intelligent Transportation Systems.* ; 2004: 743–748.
45. Achler O, Trivedi MM. Vehicle wheel detector using 2D filter banks. In: *IEEE Intelligent Vehicles Symposium.* ; 2004: 25–30.
46. Althoff M. An introduction to CORA 2015. In: *Proceedings of the Workshop on Applied Verification for Continuous and Hybrid Systems.* ; 2015.
47. Althoff M, Grebenyuk D. Implementation of interval arithmetic in CORA 2016. In: *Proceedings of the 3rd International Workshop on Applied Verification for Continuous and Hybrid Systems.* ; 2016.
48. Althoff M, Grebenyuk D, Kochdumper N. Implementation of Taylor models in CORA 2018. In: *Proceedings of the 5th International Workshop on Applied Verification for Continuous and Hybrid Systems.* ; 2018.
49. Automated parking valet toolbox. *Mathworks Inc* 2022. <https://www.mathworks.com/help/driving/ug/automated-parking-valet.html>.
50. RPLIDAR A1 Low Cost 360 Degree Laser Range Scanner: Introduction and Datasheet. *Slamtec* 2020. https://bucket-download.slamtec.com/d1e428e7efbdcd65a8ea111061794fb8d4ccd3a0/LD108_SLAMTEC_rplidar_datasheet_A1M8_v3.0_en.pdf.

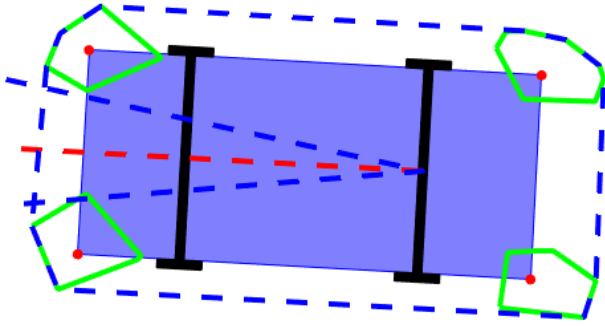
	1000 particles
	500 particles
	100 particles
	10 particles

robot body

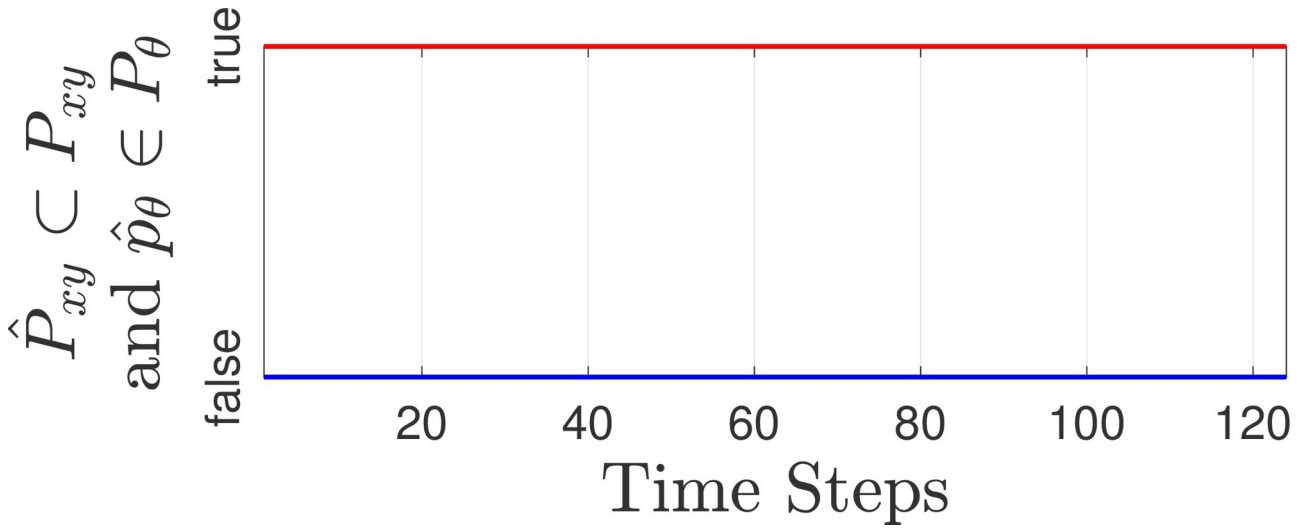


ADC2_117_collisionExample.png

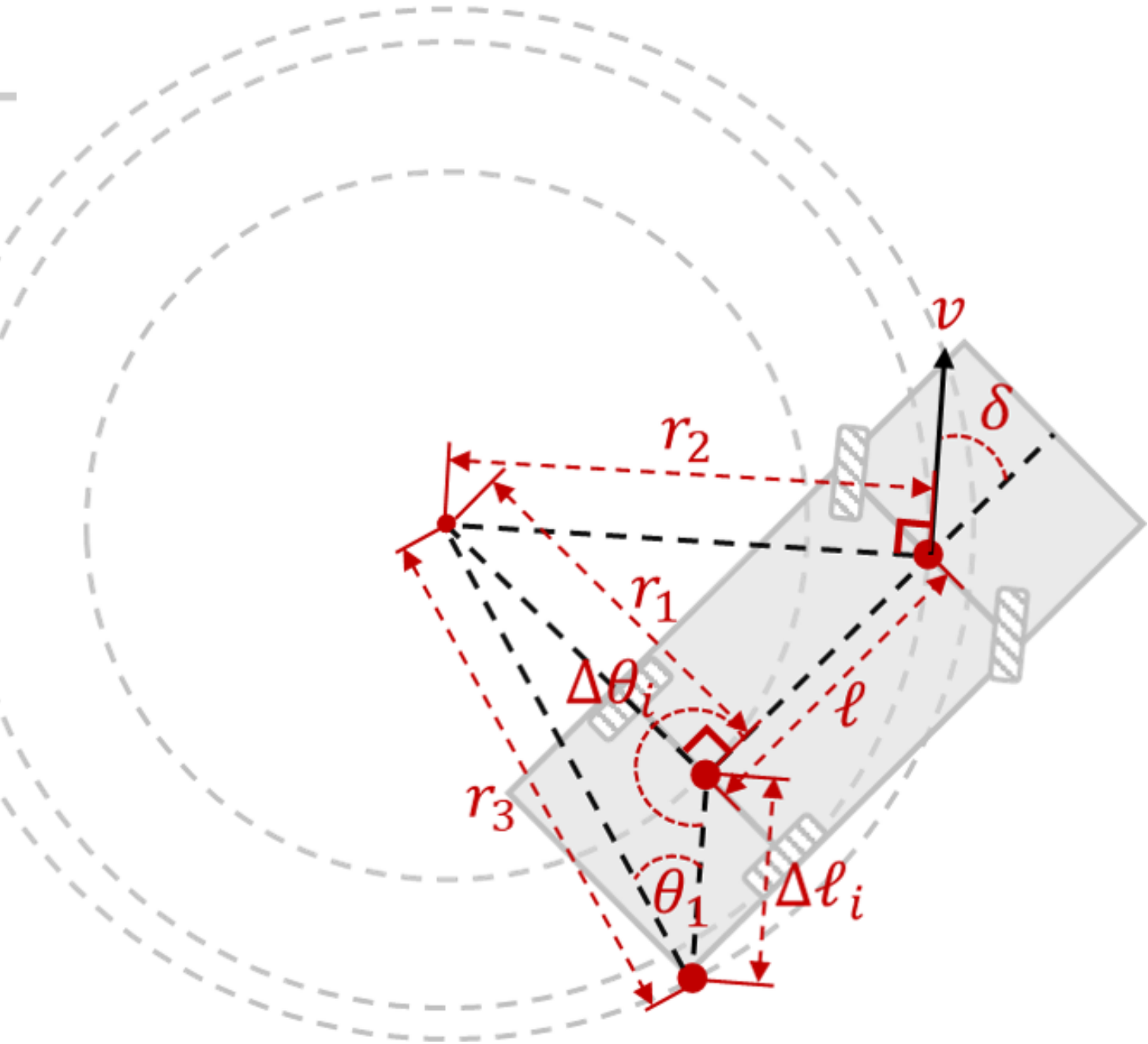
- robot body \hat{P}_{xy}
- - robot orientation \hat{p}_θ
- uncertainty set P_i of i^{th} marker
- actual position \hat{p}_i of i^{th} marker
- - estimated uncertainty sets P_{xy}, P_θ



ADC2_117_convhullRecon.png



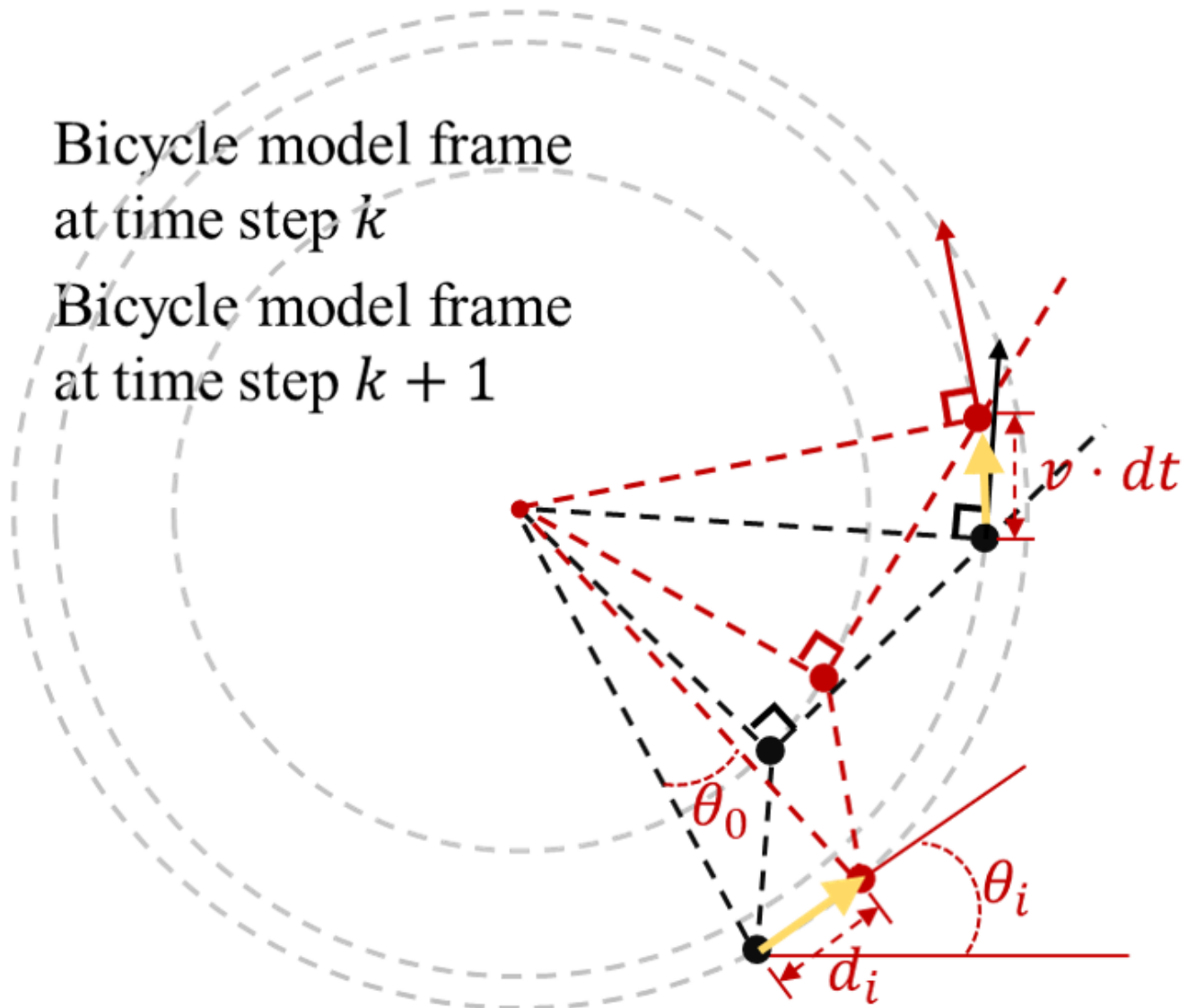
adc2_117_isinsinglerun_stereo.eps



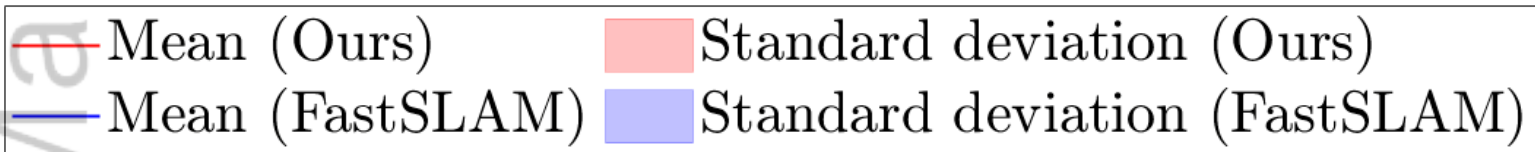
ADC2_117_kinematics_1.png

Bicycle model frame
at time step k

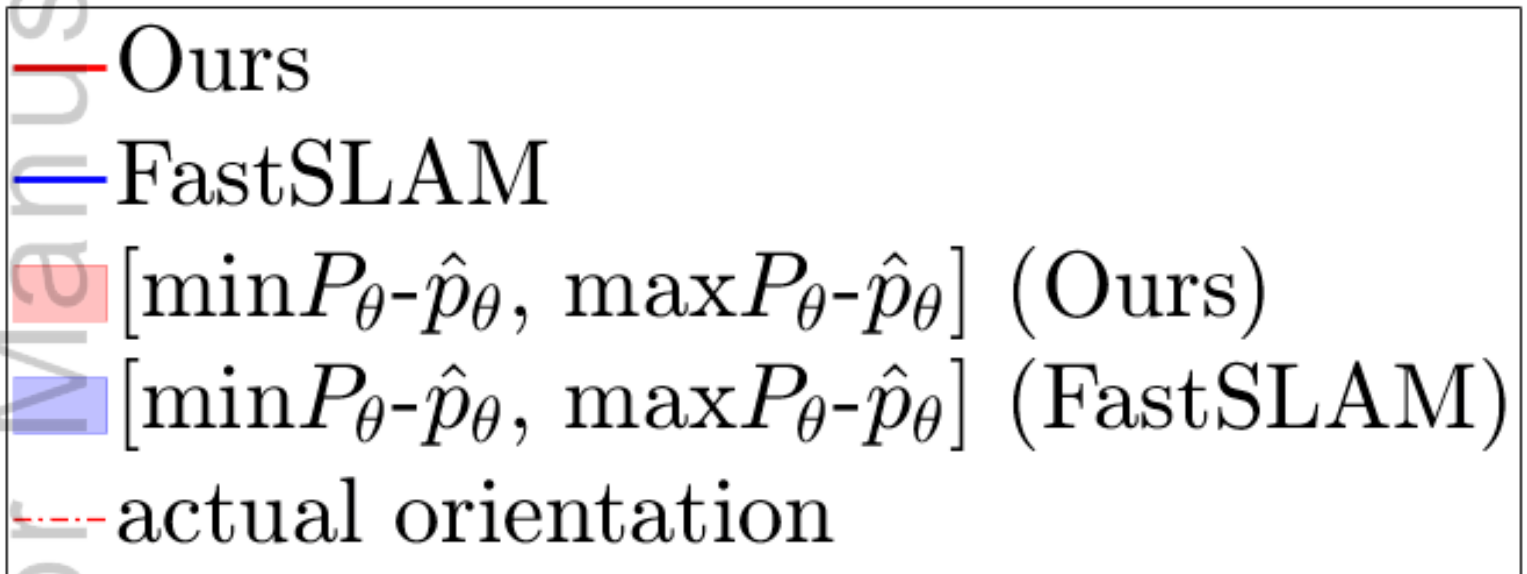
Bicycle model frame
at time step $k + 1$



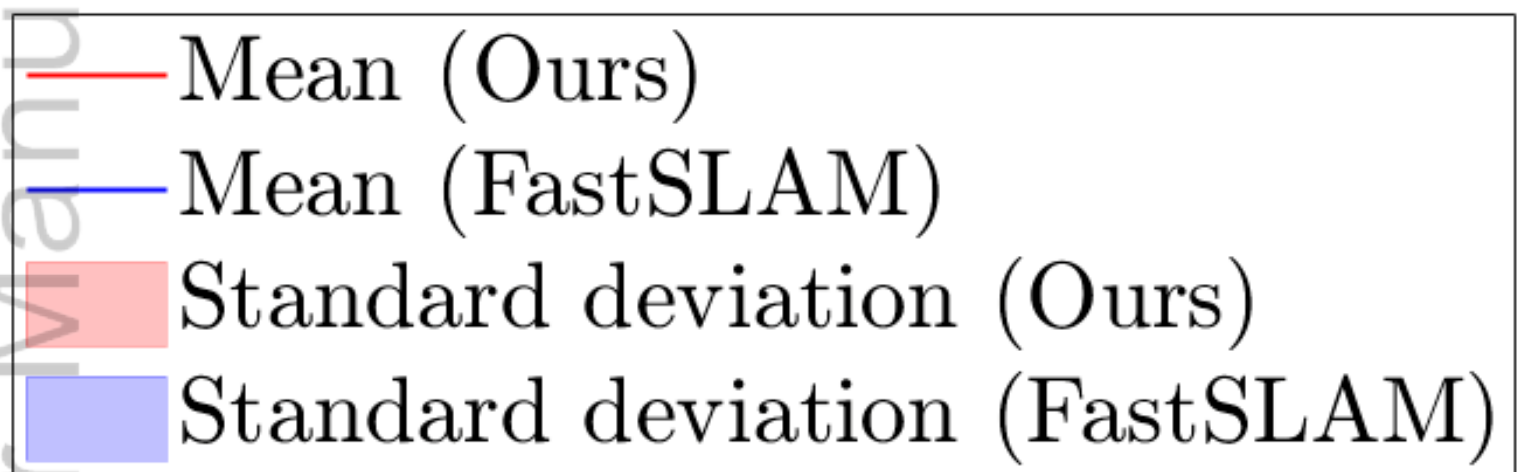
ADC2_117_kinematics_2.png



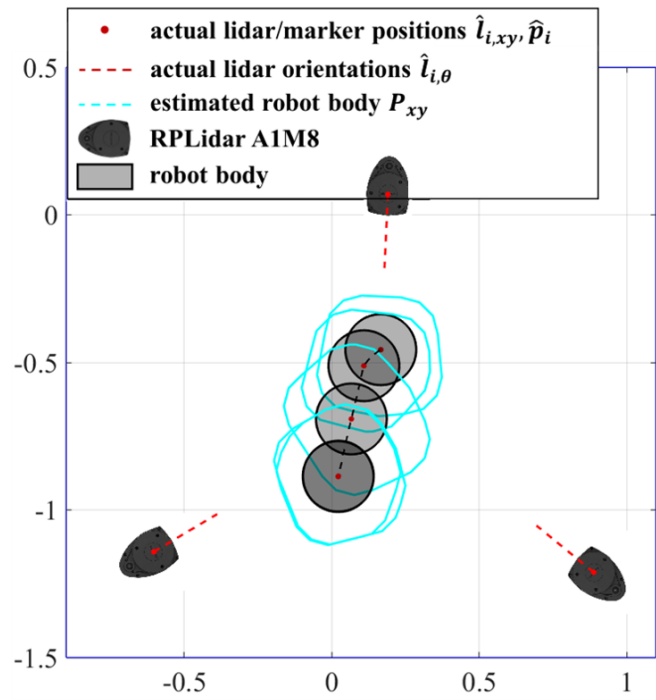
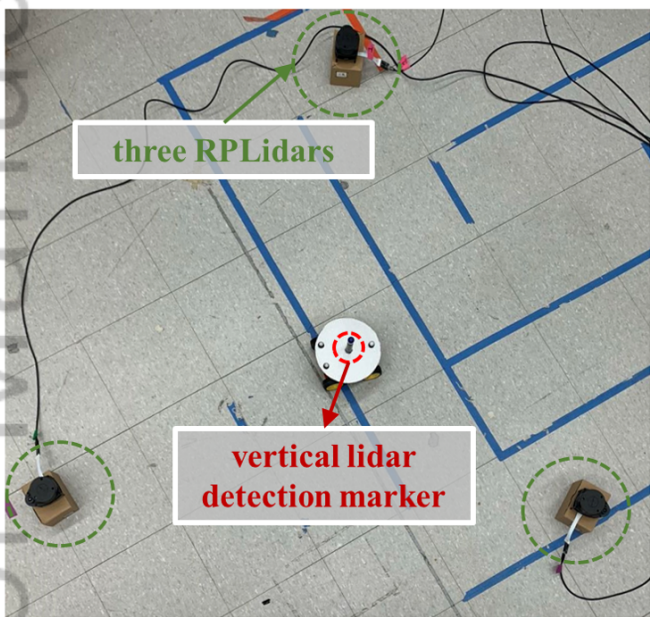
ADC2_117_legend.png



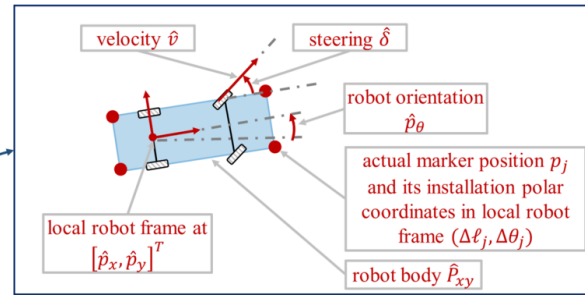
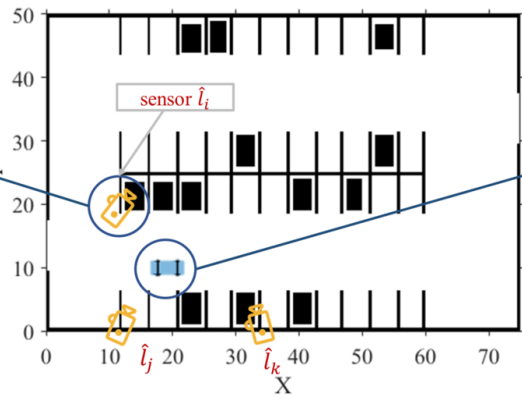
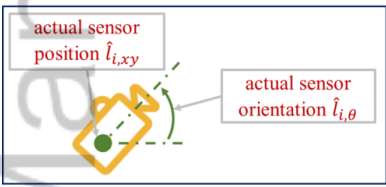
ADC2_117_legend_single.png



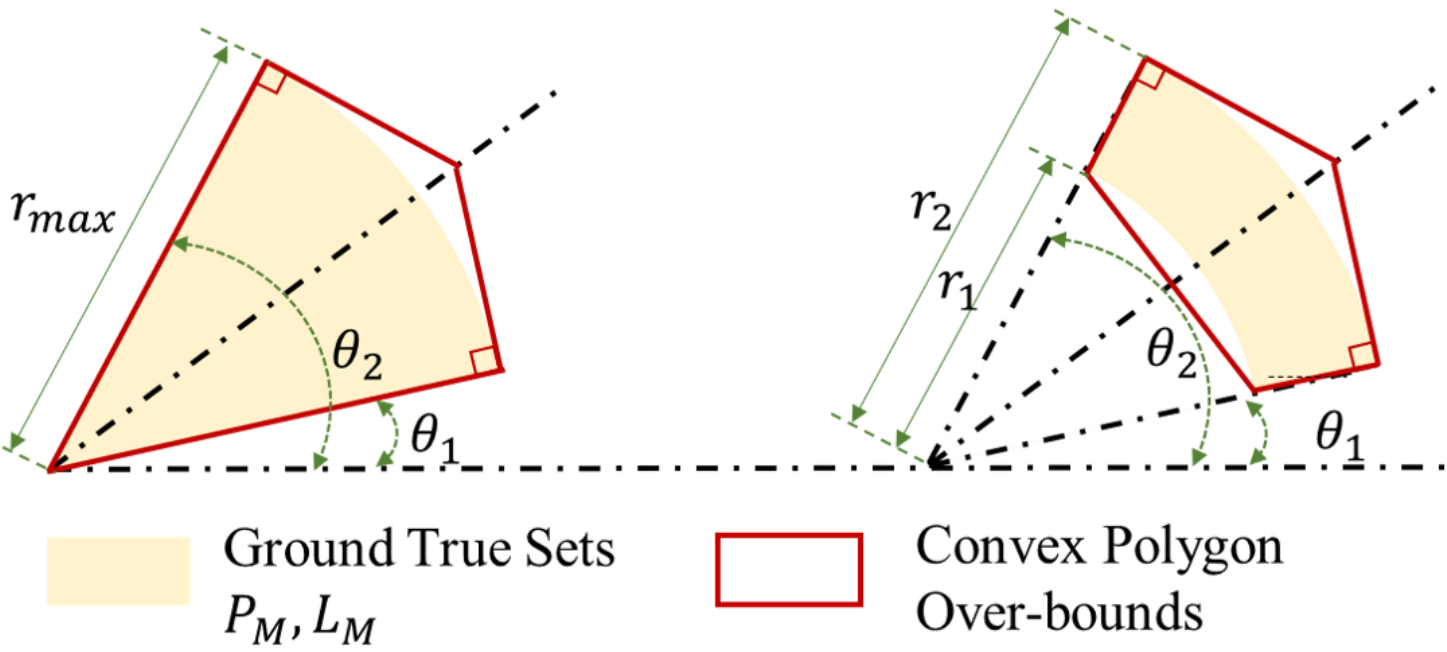
ADC2_117_legendOneCol.png



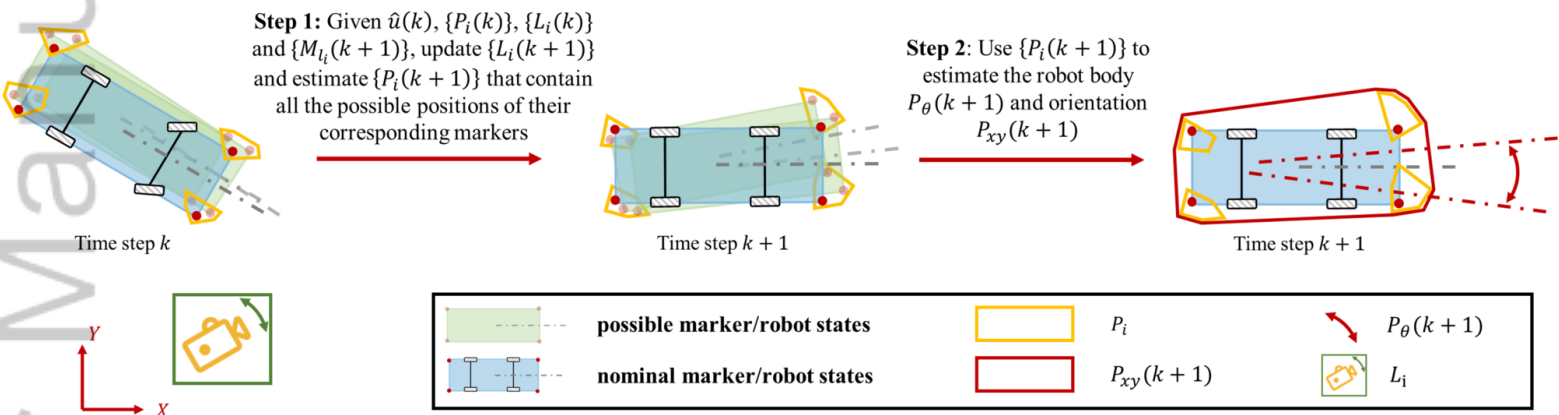
ADC2_117_lidarLoca.png



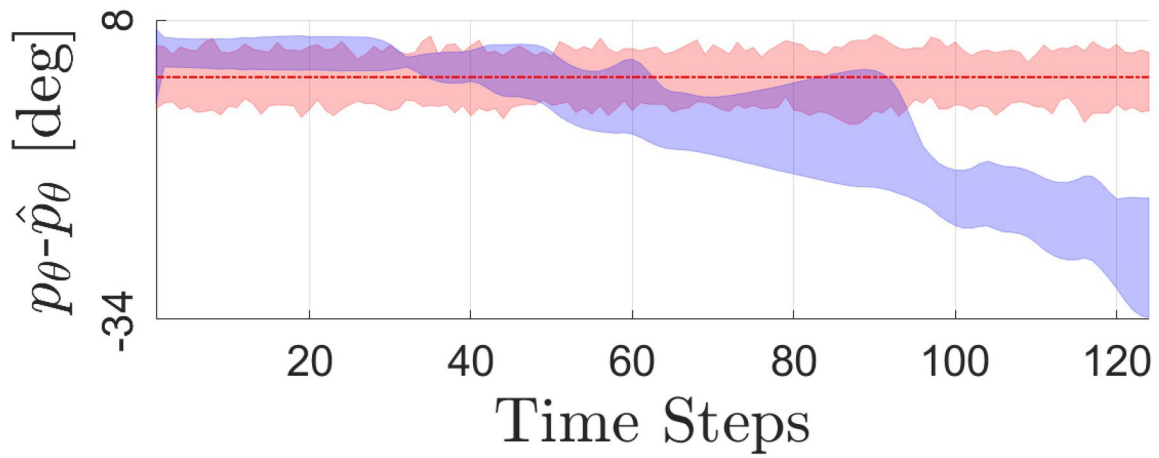
ADC2_117_parkingSpace.png



ADC2_117_polygonEst.png

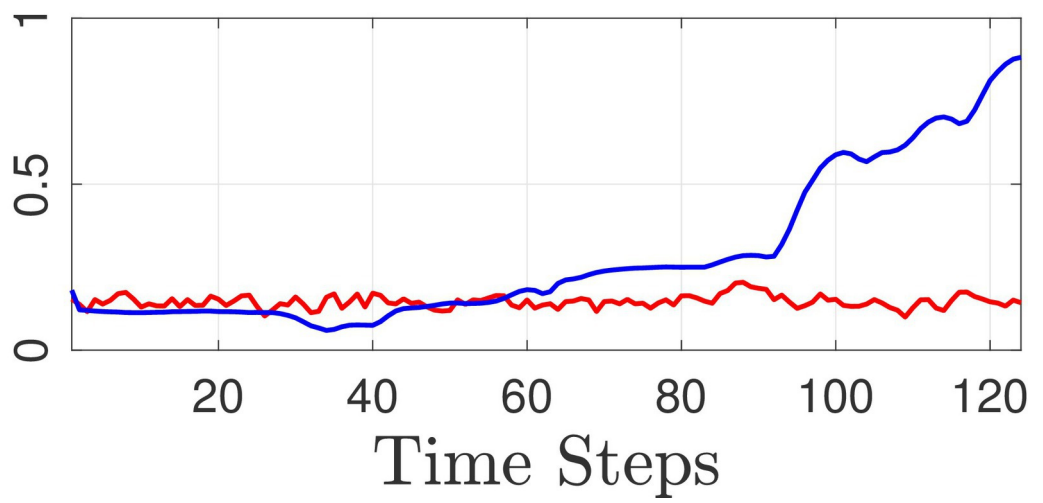


ADC2_117_problemFormulation.png

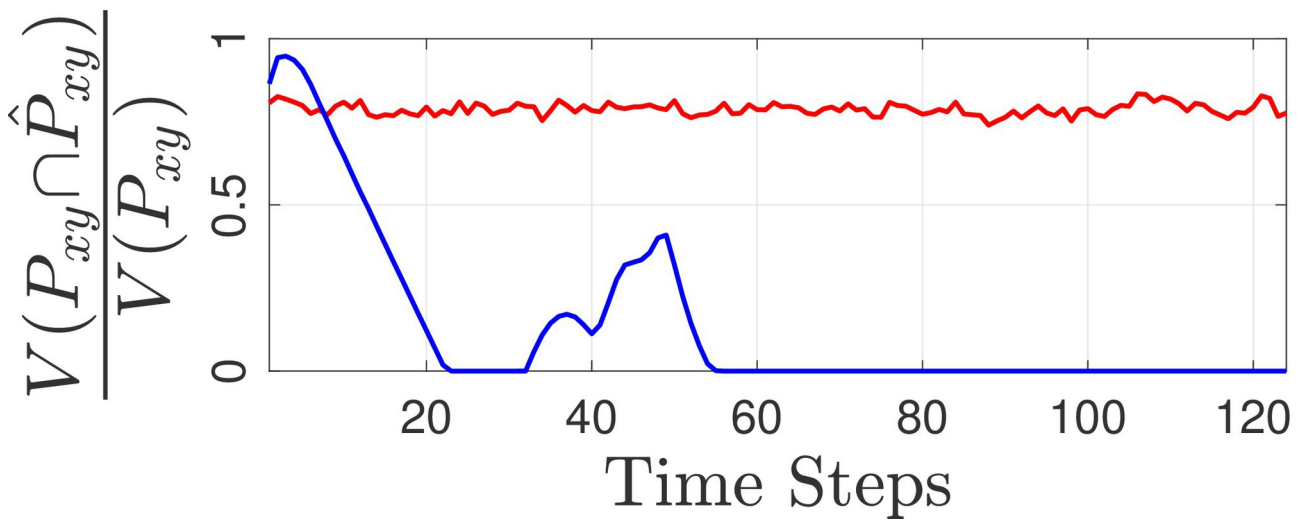


adc2_117_ptsinglerun_stereo.eps

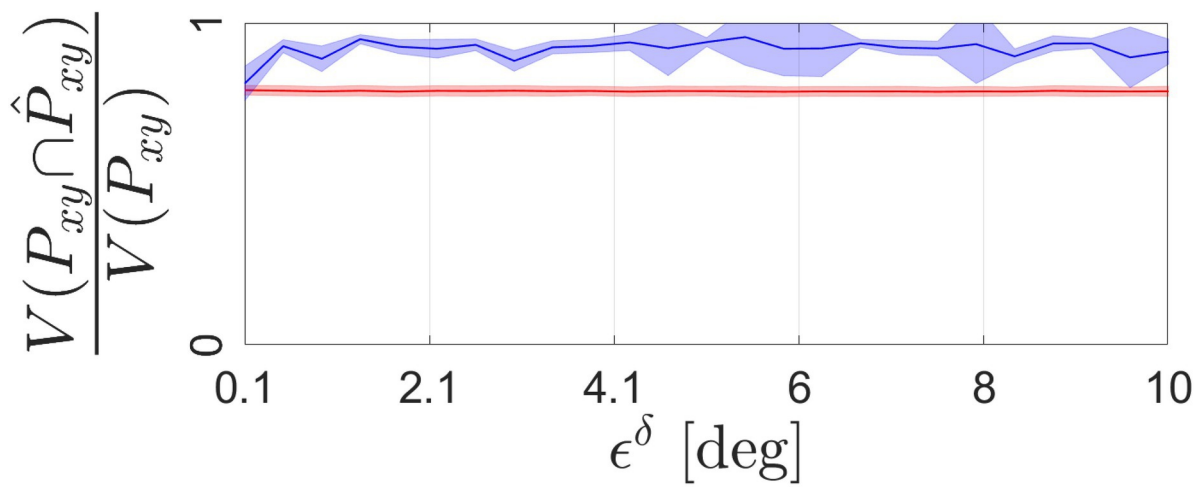
$$|\max P_\theta - \hat{p}_\theta| + |\min P_\theta - \hat{p}_\theta| \quad [\text{deg}]$$



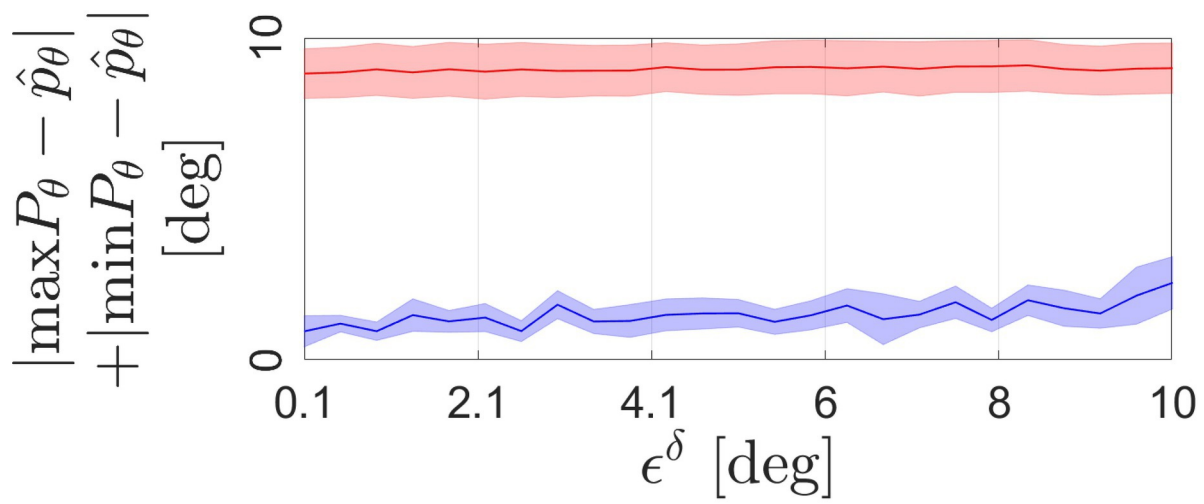
adc2_117_ptsinglerun_stereo_metric.eps



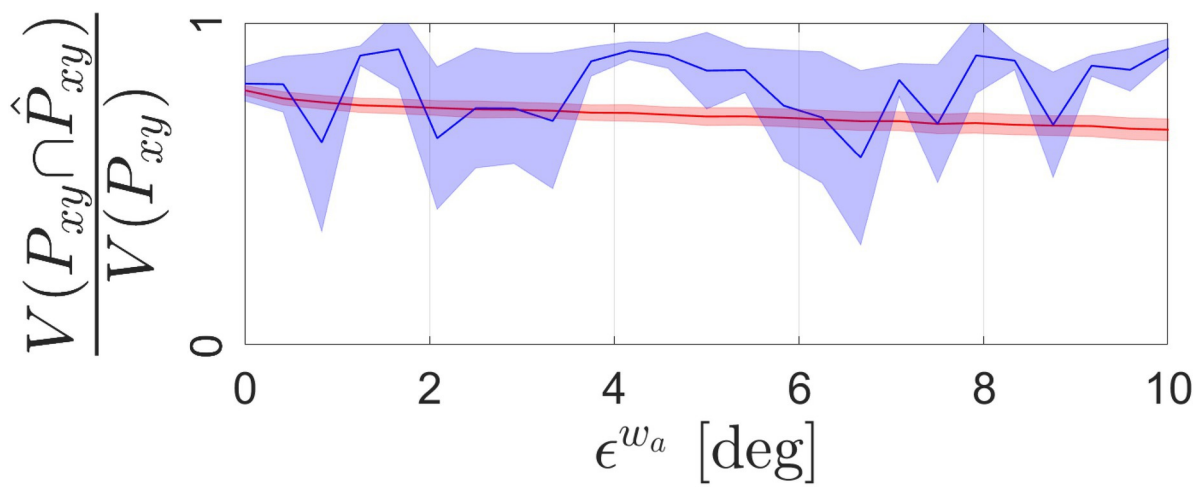
adc2_117_pxysinglerun_stereo_metric.eps



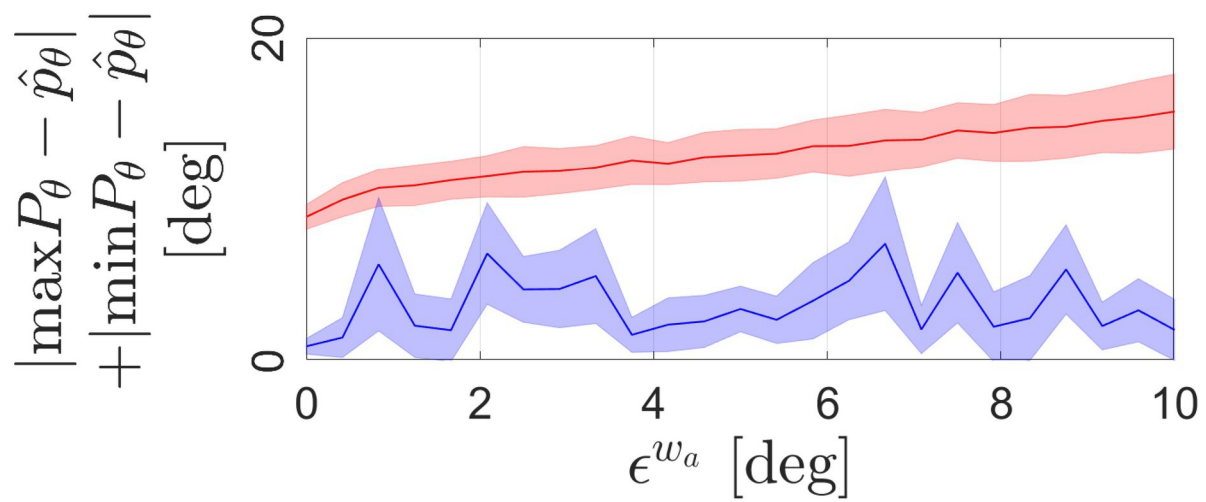
adc2_117_sensitivityunknown_stereo_delta_p.eps



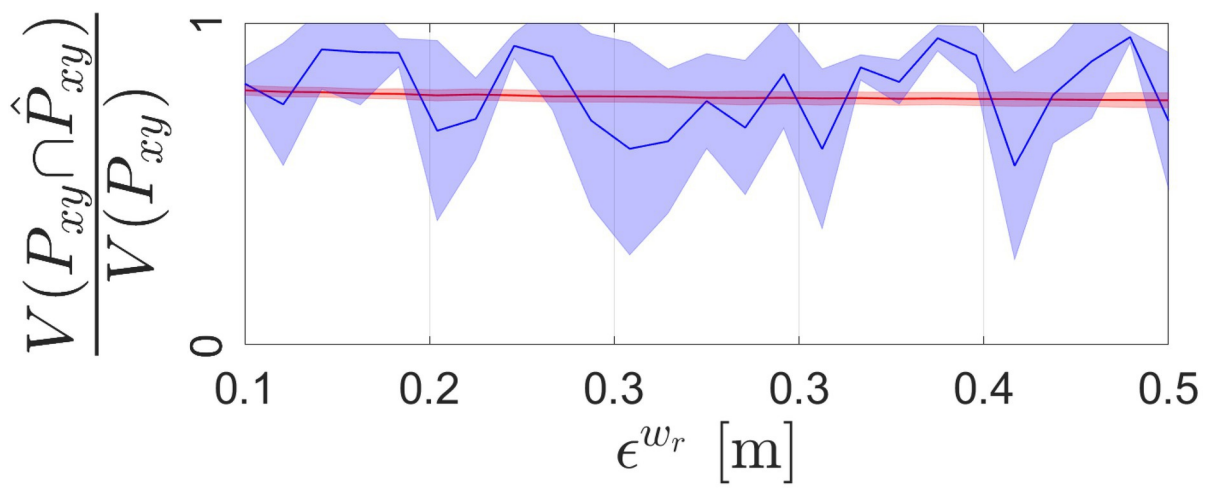
adc2_117_sensitivityunknown_stereo_delta_t.eps



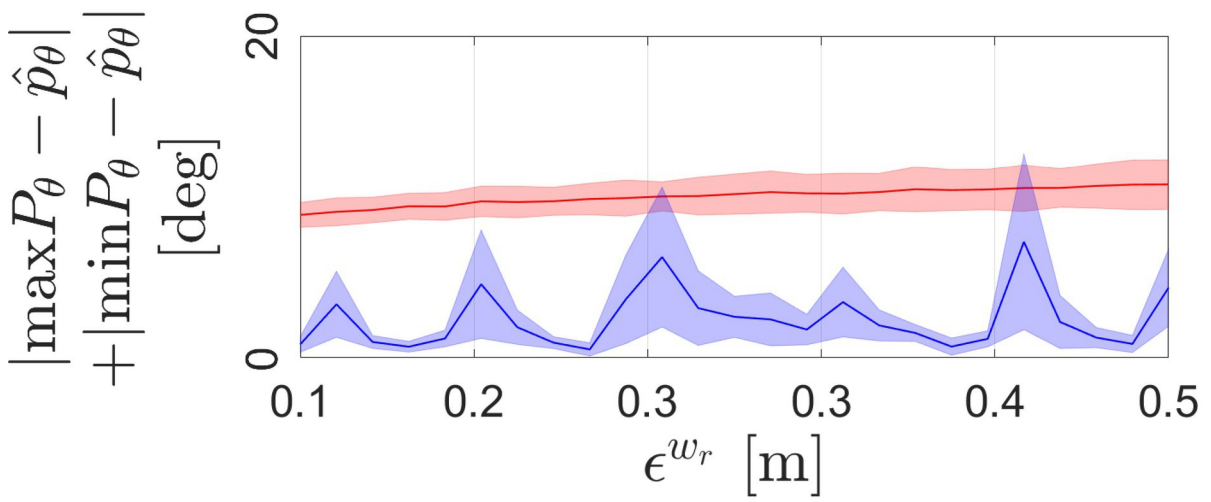
adc2_117_sensitivityunknown_stereo_ewa_p.eps



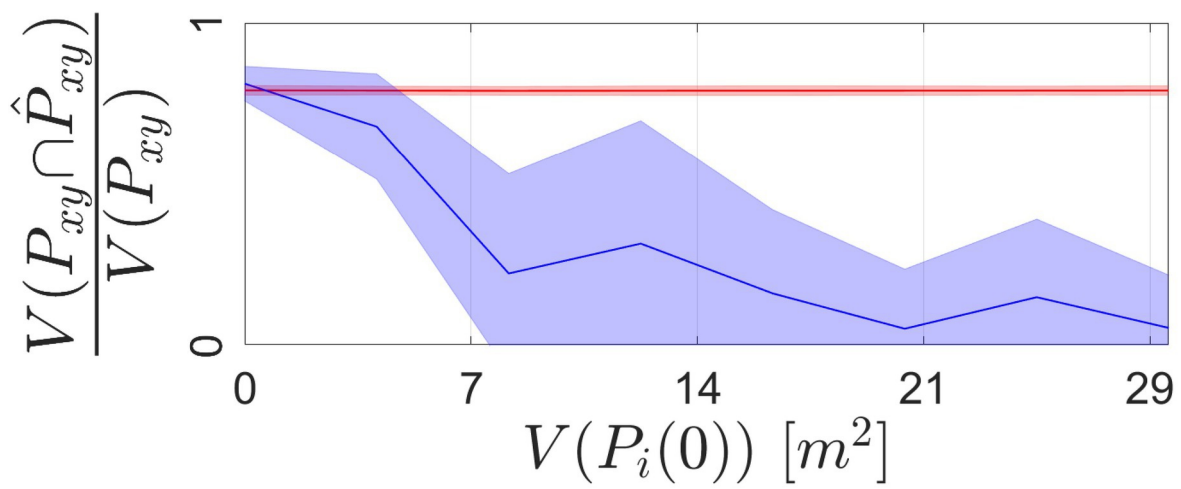
adc2_117_sensitivityunknown_stereo_ewa_t.eps



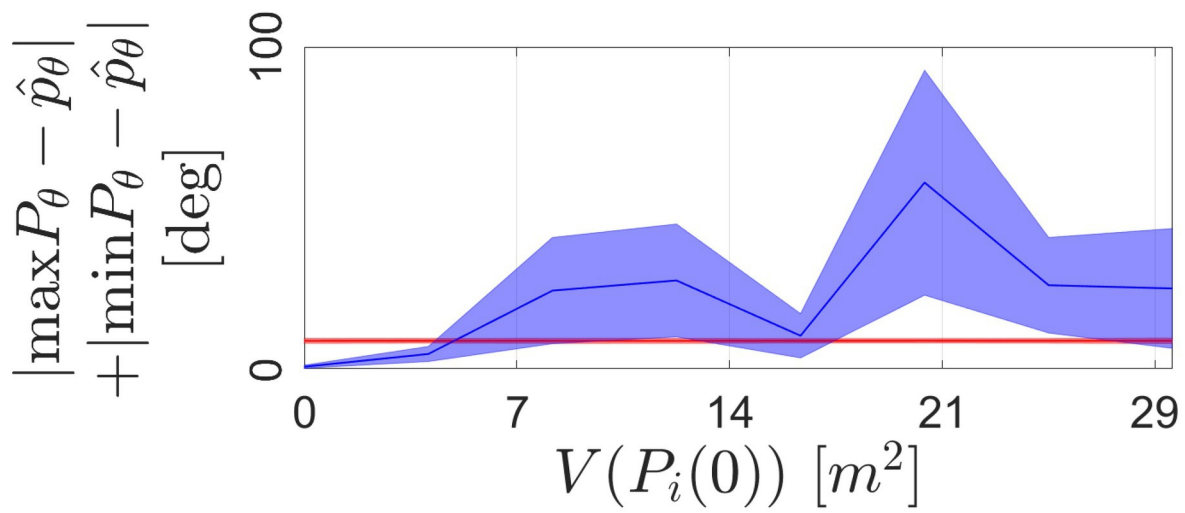
adc2_117_sensitivityunknown_stereo_ewr_p.eps



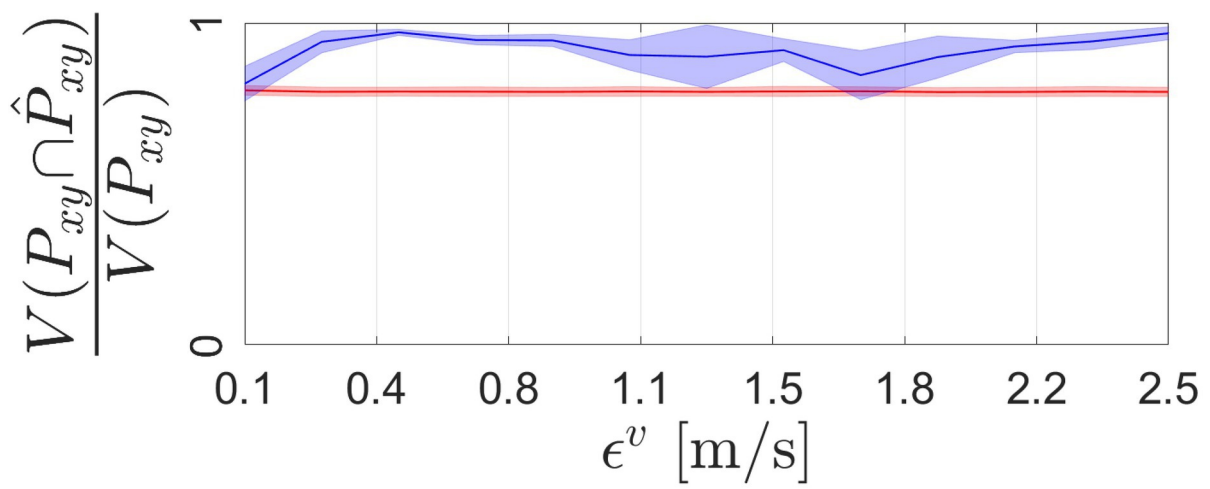
adc2_117_sensitivityunknown_stereo_ewr_t.eps



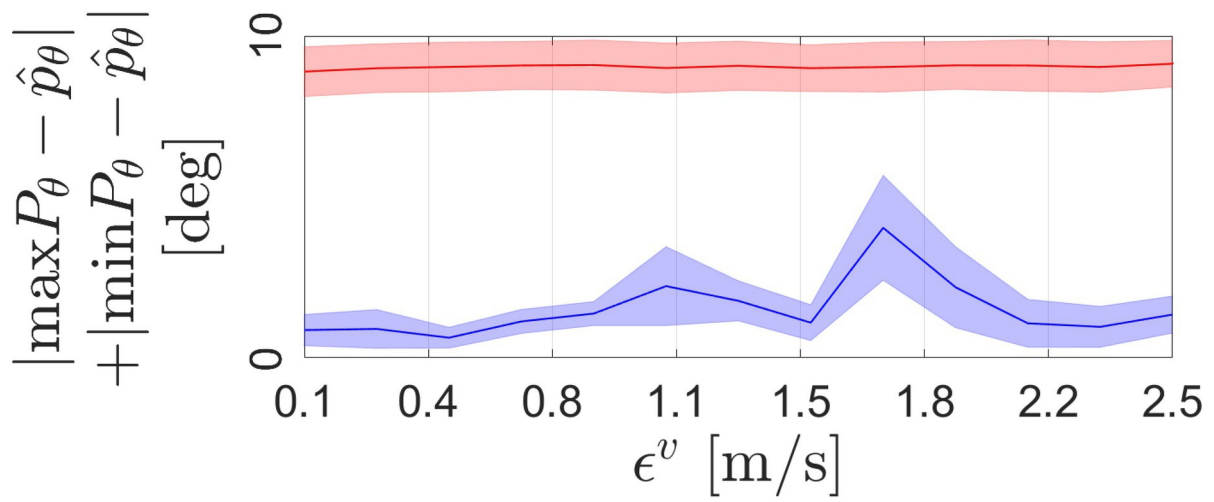
adc2_117_sensitivityunknown_stereo_pi_p.eps



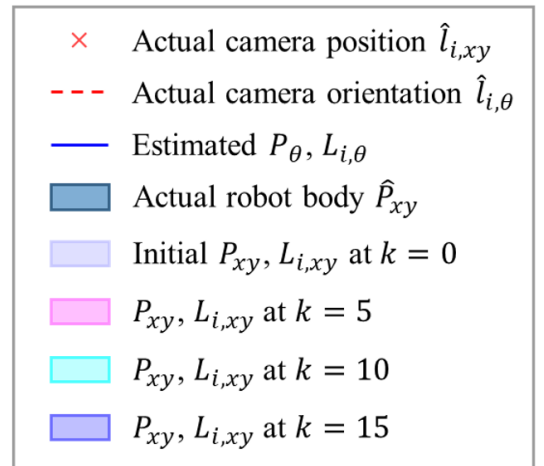
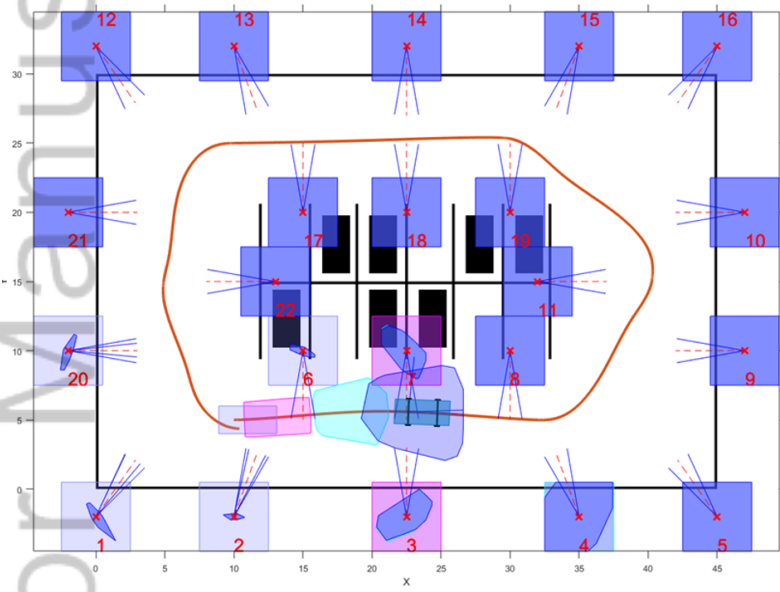
adc2_117_sensitivityunknown_stereo_pi_t.eps



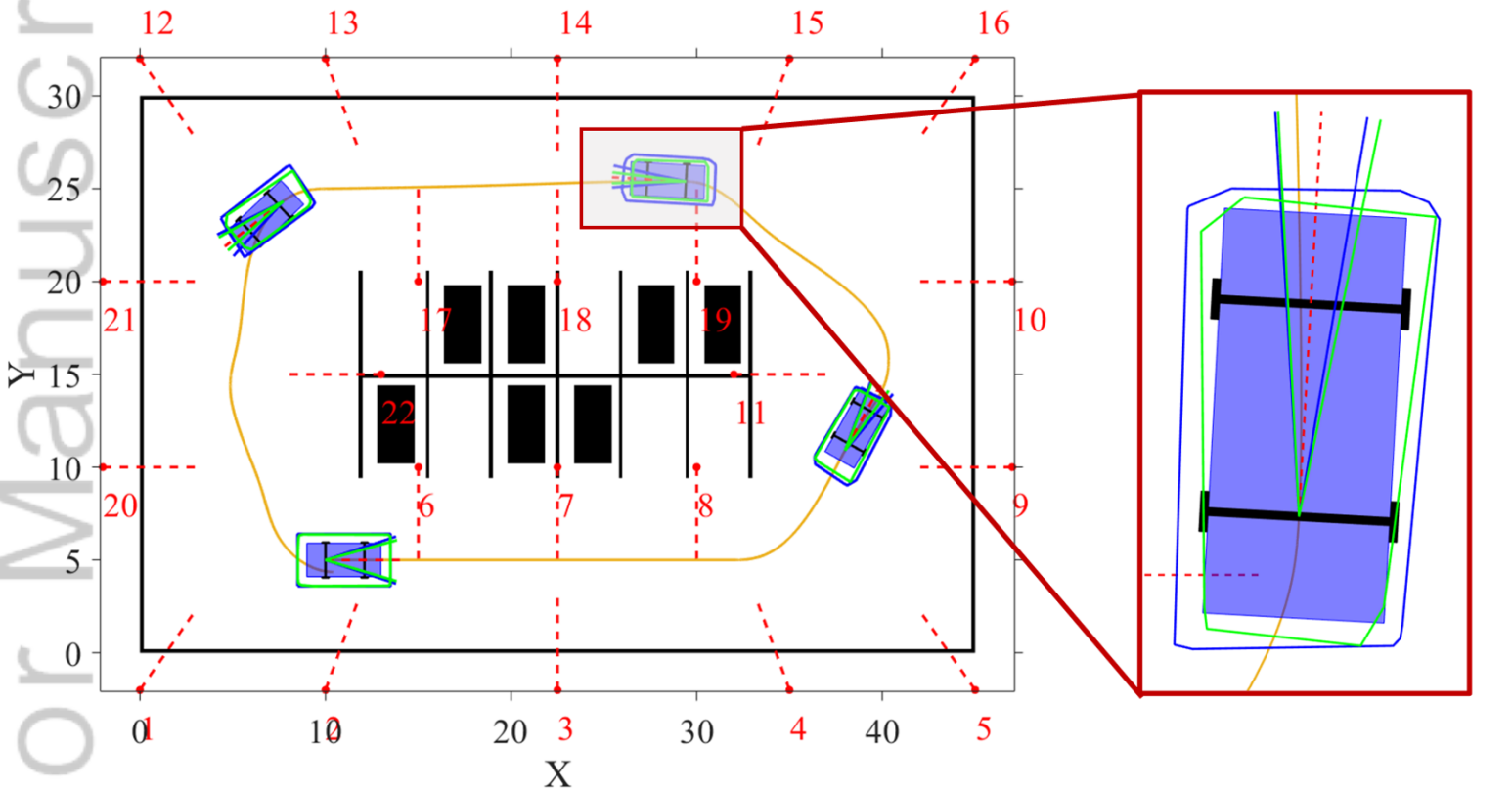
adc2_117_sensitivityunknown_stereo_v_p.eps



adc2_117_sensitivityunknown_stereo_v_t.eps



ADC2_117_sensorUpdate.png



• camera position \hat{l}_{xy}	— trajectory	— estimated P_{xy}, P_{θ} (Ours)
- - - camera/robot orientation $\hat{l}_{\theta}, \hat{p}_{\theta}$	■ trajectory snapshot	— estimated P_{xy}, P_{θ} (FastSLAM)

ADC2_117_snapshots.png