Feature-Based Perception-Aware Multi-UAV Trajectory Planning

Teaya Yang, Christian Brommer and Mark W. Mueller

***Abstract*— Multi-agent UAV systems are well-suited for large-scale data collection and transportation, though navigation in unstructured, GPS-denied environments remains challenging. Vision-based navigation enables operation without external infrastructure, but the uncertainty in state estimation limits its reliability in multi-agent settings. We propose a trajectory planning framework that incorporates estimator uncertainty by exploiting visual feature observations between agents. The framework maintains a coherent shared map through multi-agent frame alignment to prevent independent vision drift, and employs a perception-aware reward that favors trajectories with stronger feature visibility and cross-agent redundancy. Flight data from a controlled two-UAV experiment demonstrate that our alignment module can effectively reduce relative distance error, validating its role in maintaining inter-agent consistency. Simulations show that perception-aware rewards improve feature visibility and coordination while maintaining goal-reaching performance.**

I. Introduction

Uncrewed aerial vehicle (UAV) swarms provide enhanced capabilities for applications such as structural inspection, visual monitoring, and package delivery, where coordinated autonomy surpasses the performance of individual vehicles [1]. For such deployments to succeed, UAVs must maintain accurate state estimation and plan trajectories that ensure both efficiency and safety, even in GPS-denied or unstructured environments. Visual-inertial odometry (VIO) [2] has emerged as a practical solution in these scenarios, yet drift from IMU noise and feature-tracking errors remains a major challenge for reliable multi-agent operation.

Multi-agent trajectory planning for UAV systems [3]–[6] has been extensively studied in recent years, with a primary focus on enabling vehicles to avoid obstacles while coordinating and communicating within the swarm. However, most of this work assumes accurate localization and does not explicitly account for the uncertainty of vision-based estimators during flight. This limitation hinders deployment in settings where feature quality, visibility, and collectively bounded vision drifts between agents are critical to maintaining accurate mutual state estimates across agents.

In this work, we propose a trajectory planning framework that explicitly incorporates feature-based estimator uncertainty into the multi-agent planning process. Our main contributions are:

1) a trajectory planning framework that enables joint planning within a shared feature map across agents,

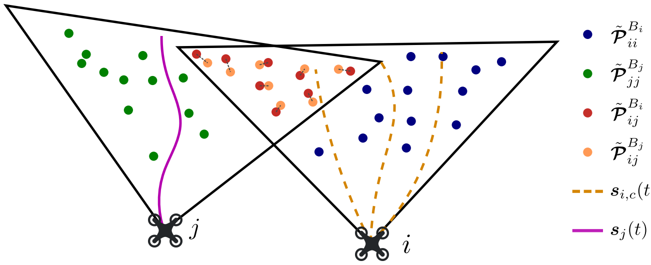

$\tilde{\mathcal{P}}_{ii}^{B_i}$
$\tilde{\mathcal{P}}_{jj}^{B_j}$
$\tilde{\mathcal{P}}_{ij}^{B_i}$
$\tilde{\mathcal{P}}_{ij}^{B_j}$
$s_{i,c}(t)$
$s_j(t)$
j
i

Fig. 1: Illustration of the multi-agent perception-aware trajectory optimization problem. The proposed framework fuses feature observations into a consistent shared map. Agent i selects its trajectory (dashed lines) from a set of candidates based on predicted information gain, while accounting for the current planned trajectory of agent j (purple line). This planning step is performed iteratively, with each agent updating its trajectory based on the latest plan of the other, resulting in joint optimization over time.

2) a feature map–based frame alignment update to mitigate independent vision drift between agents, leading to a consistent shared map for planning,
3) a perception-aware reward function that incorporates both the predicted visibility of features and the covisibility of shared features between agents.

The remainder of this paper is organized as follows. Section II reviews the relevant literature and highlights the novelty of our proposed method. Section III outlines the problem setup and assumptions. Section IV details the proposed methods, including the feature map–based frame alignment update, objective function design, and trajectory selection. Section V presents simulation results that demonstrate the effectiveness of the proposed framework.

II. Related Works

A. Vision-based navigation for UAVs

Vision-based navigation offers a robust alternative for UAV localization by avoiding dependence on external positioning infrastructure and high-cost ranging sensors, making it adaptable across different environments. For visual-inertial odometry (VIO), widely adopted methods include OpenVINS [7], which uses the multi-state constraint Kalman filter (MSCKF) [8] for computationally efficient state estimation; OKVIS [9], which performs sliding-window optimization over past states and landmarks; and VINS-Mono [10], which applies factor graph optimization to achieve high-accuracy pose estimation. For purely visual SLAM, ORB-SLAM [11] remains a popular choice due to its reliable feature tracking and loop closure capabilities. These methods, whether

The authors are with the High Performance Robotics Lab, Department of Mechanical Engineering, University of California, Berkeley, CA 94720, USA.

filtering- or optimization-based, achieve accurate localization by extracting and tracking visual features from the environment. Our approach exploits 3D feature maps for planning, which are inherently present in all of these widely used methods. In filter-based methods such as MSCKF, where 3D landmarks are not explicitly maintained in the state, they can be reconstructed from the estimated state with little computational cost.

In recent years, collaborative vision-based localization methods have been introduced for both filtering-based [12] and optimization-based approaches [13], which leverage common observations of visual features. However, these methods typically require the exchange of 2D feature tracks across agents, which can be costly in terms of communication bandwidth and imply a more centralized design. Our proposed framework introduces motion primitives that are compatible with such algorithms but supports a decentralized formulation, as it does not rely on raw data sharing or tightly coupled multi-agent localization.

B. Single-agent perception-based planning

Single-agent trajectory planning has been widely studied, and several works highlight the importance of active perception by introducing quantitative metrics for trajectory selection. Perception-aware planning for single UAVs was introduced in [14], while [15] further incorporated a model of anisotropic feature uncertainty caused by motion blur and provided empirical evidence of its effectiveness. These algorithms improve estimation robustness by steering vehicles toward feature-rich areas. More recently, [16] addressed the inverse problem, aiming to minimize mapping error rather than localization uncertainty to improve photogrammetry. Building upon the information-based trajectory costs proposed in [14], [15], our work extends perception-aware planning to multi-agent systems by explicitly modeling information gain from covisible features across agents.

In addition, we introduce a decentralized asynchronous planning framework that fuses feature maps through pairwise alignment of commonly observed features between agents, thereby coupling and bounding their vision drift, required for reliable multi-agent coordination. Our formulation enables consistent feature sharing, providing a basis for extending information-based perception-aware planning to decentralized multi-agent settings.

C. Multi-agent planning for UAVs

There has been growing progress in multi-UAV trajectory planning and the integration of sensor information [3]. Given the limitations of onboard computation, decentralized and asynchronous online planning is required for UAV swarms to operate safely and autonomously. For example, EGO-Swarm [4] is an online replanning algorithm based on depth images that enables UAVs to collaboratively avoid obstacles while efficiently advancing toward their goals. MADER [5] provides safety guarantees in decentralized, asynchronous planning, though at the cost of higher computational load. SWIFT [6] adopts a one-step learning-based approach to achieve similar capabilities with reduced complexity, making it lightweight and feasible for UAV fleet coordination. However, these algorithms do not explicitly address localization errors through planning, since trajectory evaluation is not based on estimator performance.

TABLE I: Notation summary.

Symbol	**Meaning**
$\mathcal{V} = \{1, \dots, N\}$	Set of N agents in communication.
$\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$	Set of constraints for agents with covisible features.
$\boldsymbol{\mathcal{P}}^W$	List of landmark positions observed by all agents in $\mathcal{V}$ in world frame W.
$\boldsymbol{\mathcal{P}}_i^{B_i} = \{\boldsymbol{p}_{i,1}^{B_i} \dots, \boldsymbol{p}_{i,m_i}^{B_i}\}$	List of landmark positions observed by agent i, expressed in agent i's body frame
$\boldsymbol{\mathcal{P}}_{ij}^{B_i} = \{\boldsymbol{p}_{ij,1}^{B_i} \dots, \boldsymbol{p}_{ij,m_{ij}}^{B_i}\}$	List of features observed by both agent i and j, expressed in agent i's body frame
$\boldsymbol{T}^{WB_i} = \begin{bmatrix} \boldsymbol{R}^{WB_i} & \boldsymbol{t}^{WB_i} \\ \boldsymbol{0}^\top & 1 \end{bmatrix}$	Transformation from agent i's frame to the world frame.
$\boldsymbol{T}^{B_iW}$	Transformation from the world frame to agent i's body frame.
$\boldsymbol{\xi}_i = \begin{bmatrix} \boldsymbol{\rho}_i \\ \boldsymbol{\phi}_i \end{bmatrix}$	Error in estimated vehicle pose
$\Sigma_{\mathrm{p},i}$	Covariance of pose perturbation $\boldsymbol{\xi}_i$
$\Sigma_{\mathrm{m},i}$	Covariance of 3D feature measurements in frame B_i
$S(\cdot) \in \mathbb{R}^{3\times3}$	Skew-symmetric matrix
$\boldsymbol{\Gamma}(t) = [\boldsymbol{s}(t), \boldsymbol{v}(t), \boldsymbol{a}(t)]$	Minimum-jerk trajectory primitive (position, velocity, acceleration)
$\{\boldsymbol{\Gamma}_{i,c}(t)\}_{c=1}^{n_{\mathrm{traj}}}$	Candidate trajectories generated for agent i
$\boldsymbol{s}_{G,i}$	Goal point for agent i
$\mathcal{D}_i$	Depth image taken by agent i
$\mathcal{F}_{\text{free}}$	Unoccupied space in 3D
$\mathcal{U}_{\text{feasible}}$	Set of feasible control input
$f(t)$	Normalized thrust
$\omega(t)$	Body rates
$\Lambda_{i,\tau}$	Fisher information at sampled pose τ
T	Planning duration

Few works in multi-agent trajectory planning explicitly consider localization uncertainty arising from perception quality. The most closely related work is [17], which introduces localization uncertainty into multi-agent planning. Their formulation, however, models uncertainty indirectly by propagating field-of-view–based uncertainties of detected objects, rather than leveraging the visual feature information that directly governs VIO performance. In contrast, our framework explicitly incorporates feature-tracking–induced localization uncertainty, enabling trajectory selection that proactively improves state estimation.

III. Problem Setup

We consider the perception-aware planning problem for multiple UAVs, with the objective of improving overall state estimation performance. An illustration with two agents is shown in Fig. 1. Agent i seeks to progress toward its goal point $\boldsymbol{s}_{G,i} \in \mathbb{R}^3$, where bold symbols denote vectors and the subscript G, i indicates the goal position of agent i in the world frame. During flight, the agent performs state estimation using vision-based methods such as those outlined in Section II-A. These methods output an estimated pose $\tilde{\boldsymbol{T}}_i^{WB_i} \in SE(3)$ with its associated covariance, where the tilde denotes an estimated (rather than true) quantity. Here, W denotes the world frame and B_i denotes the body frame of agent i. For simplicity, we omit the intermediate transformation between each body and its onboard camera frame. Each agent also maintains a local 3D feature map $\tilde{\boldsymbol{\mathcal{P}}}_i^{B_i} = \{\tilde{\boldsymbol{p}}_{i,1}^{B_i}, \ldots, \tilde{\boldsymbol{p}}_{i,m_i}^{B_i}\}$ expressed in its body frame B_i and reconstructed from a short history of visual observations. At discrete communication times, these estimated quantities can be shared across agents.

Upon receiving shared information, each agent performs a feature-based alignment step (Section IV-A), by fusing received features into a unified map which anchors the relative localization between agents, leading to a locally consistent and shared vision drift. The planning problem is then to select a trajectory $\boldsymbol{\Gamma}_i(t) = [\boldsymbol{s}_i(t), \boldsymbol{v}_i(t), \boldsymbol{a}_i(t)]$ that advances toward $\boldsymbol{s}_{G,i}$ while enhancing expected localization quality, subject to safety and input feasibility. This problem is formulated as

$$\begin{aligned} \arg\max_{\boldsymbol{\Gamma}_i(t)} \quad & k_{\text{goal}} R_{\text{goal},i} + k_{\text{perc}} R_{\text{perc},i} && (1) \\ \text{s.t.} \quad & \boldsymbol{s}_i(t) \in \mathcal{X}_{\text{free}}, \quad \forall t \in [0, T], && (2) \\ & f_i(t), \omega_i(t) \in \mathcal{U}_{\text{feasible}}, \quad \forall t \in [0, T]. && (3) \end{aligned}$$

The objective (1) combines the goal progress reward $R_{\text{goal},i}$ and perception-aware reward $R_{\text{perc},i}$, weighted by user-defined scalar weights k_{goal} and k_{perc}. Constraint (2) enforces that the trajectory remains in the collision-free space $\mathcal{X}_{\text{free}}$, and (3) enforces that the normalized thrust $f_i(t)$ and body rates $\omega_i(t)$ associated with the candidate trajectory lie within the feasible input set $\mathcal{U}_{\text{feasible}}$. Each trajectory is defined over a finite time horizon $0 < t < T$, where $T > 0$ denotes the trajectory duration. The value of T may be fixed or sampled during trajectory generation.

This problem can be solved in real time via a sampling-based approach (Section IV-B). Candidate trajectories are evaluated using (1), with the computation of the reward terms detailed in Section IV-C, and the highest-reward trajectory satisfying the constraints is selected. To evaluate R_{perc}, we sample a finite set of poses along each candidate trajectory and assume that the latest planned trajectories of neighboring agents are known through communication, enabling prediction of future visible and covisible features. Shared trajectories could also be leveraged for formation or coordination objectives, which are beyond the scope of the present work. Fig. 2 illustrates the overall system architecture, highlighting the data flow and key modules.

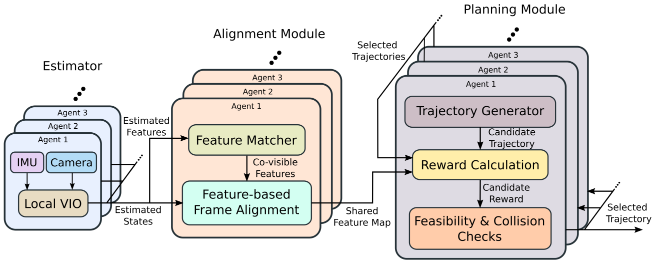


Fig. 2: Software architecture of the proposed framework. Each agent runs a local VIO estimator to generate states and visual features. These are shared among agents and covisible features are aligned producing a unified map. The planner then uses this map with trajectory information to sample candidates, evaluate rewards, check feasibility, and select the best trajectory.

IV. Proposed Planning Framework

A. Feature-based frame alignment

Evaluating the cost in (1) requires access to $\boldsymbol{\mathcal{P}}^W$, a feature map in the consensus world frame containing unique landmarks. To enable perception-aware planning, this map must be constructed by fusing observations shared among all agents in the communication set $\mathcal{V}$. We assume that the system is time-synchronized, so that the covisible features being fused are observed at the same instant across agents. This section introduces the proposed alignment algorithm, which formulates the fusion as an optimization problem that enforces pairwise constraints from these covisible features.

Given the sets of features observed by two communicating agents i and j, we first identify the commonly observed features using descriptor matching. For each such pair $(i, j) \in \mathcal{E}$, this partitions the features into four sets: $\tilde{\boldsymbol{\mathcal{P}}}_{ii}^{B_i}$ and $\tilde{\boldsymbol{\mathcal{P}}}_{jj}^{B_j}$, which are unique to each agent, and $\tilde{\boldsymbol{\mathcal{P}}}_{ij}^{B_i}$ and $\tilde{\boldsymbol{\mathcal{P}}}_{ij}^{B_j}$, which correspond to shared features measured independently by both agents. The number of such covisible features between the pair is denoted m_{ij}. The map-based frame alignment then solves for the updated vehicle poses together with the positions of these shared features, using the pairwise constraints. For each communicating pair of agents with covisible features, we jointly refine their poses and the shared landmarks by solving a nonlinear least-squares problem. This takes the standard form used in landmark-based SLAM [18], where residuals capture deviations from pose priors and feature measurement consistency. The resulting optimization problem is

$$\min_{\Theta} \sum_{i \in \mathcal{V}} \|r_{\text{p},i}\|^2_{\Sigma^{-1}_{\text{p},i}} + \sum_{(i,j) \in \mathcal{E}} \sum_{k=1}^{m_{ij}} \left(\|r_{\text{m},i,k}\|^2_{\Sigma^{-1}_{\text{m},i}} + \|r_{\text{m},j,k}\|^2_{\Sigma^{-1}_{\text{m},j}} \right), \tag{4}$$

where the first summation penalizes the pose prior residuals $r_{\text{p},i}$ for each agent $i \in \mathcal{V}$, weighted by the inverse of the pose covariance $\Sigma_{\text{p},i}$ provided by the VIO estimator. The second term accumulates the measurement residuals $r_{\text{m},i,k}$ and $r_{\text{m},j,k}$ for each covisible feature k observed by a communicating agent pair $(i, j) \in \mathcal{E}$, weighted by the corresponding feature measurement covariances $\Sigma_{\text{m},i}$ and $\Sigma_{\text{m},j}$. The notation $\|r\|^2_{\Sigma^{-1}} := r^\top \Sigma^{-1} r$ denotes the squared Mahalanobis norm.

The decision variable is defined as

$$\Theta = \{\boldsymbol{\xi}_i\}_{i\in\mathcal{V}} \;\cup\; \{\boldsymbol{\mathcal{P}}^W_{ij,k}\}_{(i,j)\in\mathcal{E},\, k=1..m_{ij}}, \tag{5}$$

which consists of the pose perturbations $\boldsymbol{\xi}_i \in \mathbb{R}^6$ for each agent $i \in \mathcal{V}$, parameterizing corrections to the estimated pose $\tilde{\boldsymbol{T}}^{WB_i}_i$, and the 3D positions $\boldsymbol{\mathcal{P}}^W_{ij,k}$ of landmarks k observed by both agents i and j, expressed in the consensus world frame. The updated pose is obtained by applying a left perturbation to the estimate,

$$\boldsymbol{T}^{WB_i} = \exp(\boldsymbol{\xi}_i)\,\tilde{\boldsymbol{T}}^{WB_i},$$

where $\exp(\cdot)$ denotes the matrix exponential mapping from $\mathfrak{se}(3)$ to $\mathrm{SE}(3)$.

Since $\boldsymbol{\xi}_i = \boldsymbol{0}$ when the pose error is zero, the prior residual for agent i is defined as

$$r_{\mathrm{p},i} = \boldsymbol{\xi}_i. \tag{6}$$

The measurement residuals for a covisible feature k between agents i and j are

$$\begin{aligned} r_{\mathrm{m},i,k} &= \boldsymbol{R}^{B_iW}\boldsymbol{p}^W_{ij,k} + \boldsymbol{t}^{B_iW} - \tilde{\boldsymbol{p}}^{B_i}_{ij,k}, \\ r_{\mathrm{m},j,k} &= \boldsymbol{R}^{B_jW}\boldsymbol{p}^W_{ij,k} + \boldsymbol{t}^{B_jW} - \tilde{\boldsymbol{p}}^{B_j}_{ij,k}, \end{aligned} \tag{7}$$

where $\boldsymbol{R}^{B_iW}$ and $\boldsymbol{t}^{B_iW}$ transform world-frame points into body frame B_i, and $\tilde{\boldsymbol{p}}^{B_i}_{ij,k}$ denotes the observed position in B_i. Linearizing about the current estimated pose,

$$r_{\mathrm{m},i,k} \approx \tilde{\boldsymbol{R}}^{B_iW}\boldsymbol{p}^W_{ij,k} + \tilde{\boldsymbol{t}}^{B_iW} + J^p_{\xi_i,k}\,\boldsymbol{\xi}_i - \tilde{\boldsymbol{p}}^{B_i}_{ij,k},$$

$$J^p_{\xi_i,k} := \left.\frac{\partial \boldsymbol{p}^{B_i}_{ij,k}}{\partial \boldsymbol{\xi}_i}\right|_{\boldsymbol{\xi}_i=0} = -\tilde{\boldsymbol{R}}^{B_iW}\left[\boldsymbol{I}_{3\times3} \quad -S(\boldsymbol{p}^W_{ij,k})\right]. \tag{8}$$

The residual for agent j is defined analogously.

Solving (4) with decision variables in (5) yields corrected poses for all communicating agents and a fused map of covisible features, $\{\boldsymbol{\mathcal{P}}^W_{ij}\}_{(i,j)\in\mathcal{E}}$, expressed in the consensus world frame. The singly observed features can also be transformed into this frame using the corrected poses using

$$\boldsymbol{\mathcal{P}}^W_{ii} = \boldsymbol{R}^{WB_i}\,\tilde{\boldsymbol{\mathcal{P}}}^{B_i}_{ii} + \boldsymbol{t}^{WB_i}, \qquad \forall i \in \mathcal{V}.$$

The shared feature map used for planning is then obtained as the union

$$\boldsymbol{\mathcal{P}}^W = \left(\bigcup_{(i,j)\in\mathcal{E}} \boldsymbol{\mathcal{P}}^W_{ij}\right) \cup \left(\bigcup_{i\in\mathcal{V}} \boldsymbol{\mathcal{P}}^W_{ii}\right).$$

B. Trajectory optimization

We adopt a trajectory selection framework similar to [15], where candidate minimum-jerk trajectories are generated following [19]. The procedure is summarized in Algorithm 1, where each candidate trajectory is scored using (1), with the lowest-cost feasible, collision-free trajectory selected. Collision avoidance can be achieved with onboard perception algorithms such as [20].

At each planning step for agent i, we assume that the latest planned trajectories of all other agents $j \in \mathcal{V}\setminus\{i\}$ are available through communication. A set of n_{traj} candidate trajectories for agent i is generated, denoted $\{\boldsymbol{\Gamma}_{i,c}(t)\}_{c=1}^{n_{\text{traj}}}$. Each trajectory $\boldsymbol{\Gamma}_{i,c}(t)$ is discretized into a sequence of predicted poses $\{\boldsymbol{T}^{WB_i}_{c,\tau}\}_{\tau=1}^{n_{\text{poses}}}$. Similarly, the communicated trajectories $\boldsymbol{\Gamma}_j(t)$ from other agents are sampled into $\{\boldsymbol{T}^{WB_j}_{\tau}\}_{\tau=1}^{n_{\text{poses}}}$. Here, c indexes candidate trajectories and τ indexes the discretized poses along each trajectory. These pose sequences are then used to compute the perception-based reward R_{perc} described in Section IV-C.

Algorithm 1 Sampling-Based Trajectory Optimization

```
input: $\mathcal{D}_i$, $\tilde{\boldsymbol{T}}^{BW_i}$, $\boldsymbol{\Gamma}_i(0)$, $\boldsymbol{\mathcal{P}}^W$, $\{\boldsymbol{\Gamma}_j(t)\}_{j\in\mathcal{V}\setminus\{i\}}$, $\boldsymbol{s}_{G,i}$
output: $\boldsymbol{\Gamma}^*_i(t)$, or undefined (failure)
function FindHighestRewardTrajectory
    $\boldsymbol{\Gamma}^*_i(t) \leftarrow$ undefined with $\mathrm{Reward}(\boldsymbol{\Gamma}^*_i(t)) = -\infty$
    while computation time not exceeded do
        $\boldsymbol{\Gamma}_{i,c}(t) \leftarrow$ GetCandidateTraj [19]
        if $\mathrm{Reward}(\boldsymbol{\Gamma}_{i,c}(t)) > \mathrm{Reward}(\boldsymbol{\Gamma}^*_i(t))$ then
            if IsDynamicallyFeas($\boldsymbol{\Gamma}_{i,c}(t)$) then
                if IsCollisionFree($\boldsymbol{\Gamma}_{i,c}(t), \mathcal{D}_i$) then
                    $\boldsymbol{\Gamma}^*_i(t) \leftarrow \boldsymbol{\Gamma}_{i,c}(t)$
                end if
            end if
        end if
    end while
    return $\boldsymbol{\Gamma}^*_i(t)$
end function
```

C. Trajectory reward

1) Goal progress reward: The goal progress reward encourages the agent to reduce its distance to the target as efficiently as possible. Following [15], we define

$$R_{\text{goal},i} = \frac{\|\boldsymbol{s}_{G,i} - \boldsymbol{s}(0)\| - \|\boldsymbol{s}_{G,i} - \boldsymbol{s}(T)\|}{T},$$

where the numerator represents the decrease in distance to the goal over the trajectory, and the division by T normalizes this reduction by the trajectory duration. A higher reward is obtained when the agent moves closer to its goal at a faster rate, while trajectories that deviate from the goal or make little progress yield lower values.

2) Information-based perception reward: The perception reward captures the expected information gain from visual observations, thereby encouraging trajectories that improve localization accuracy. We formulate this reward using a Fisher information-based framework [21], which quantifies how feature visibility and inter-agent covisibility reduce state uncertainty. At each future time step, assuming the agents solve the problem in (4), two sources of information are considered: (i) the contribution of independent VIO output, obtained by tracking a history of 2D features, and (ii) the contribution of covisible 3D feature measurements shared between agents, represented by the measurement terms.

The prior-related information gain follows the formulation in [15], which approximates the contribution of feature tracks to pose refinement. At a sampled pose τ of agent i, let m_τ

denote the number of features visible in the camera. Each feature k has a corresponding 2D measurement $\tilde{\boldsymbol{b}}_{i,\tau,k}$ and a 3D world position $\boldsymbol{p}_k^W$. Expressed in the body frame B_i, the feature position is

$$\boldsymbol{p}_{i,\tau,k}^{B_i} = [x_{i,\tau,k}^{B_i},\, y_{i,\tau,k}^{B_i},\, z_{i,\tau,k}^{B_i}]^\top.$$

The projection model then gives

$$\boldsymbol{b}_{i,\tau,k} = \left[f_x \frac{x_{i,\tau,k}^{B_i}}{z_{i,\tau,k}^{B_i}} + c_x \quad f_y \frac{y_{i,\tau,k}^{B_i}}{z_{i,\tau,k}^{B_i}} + c_y \right]^\top,$$

where f_x and f_y are focal lengths and c_x and c_y are the principal point offsets. The estimated body pose at this time is given by $\tilde{\boldsymbol{T}}_\tau^{B_i W}$, and we introduce a perturbation $\boldsymbol{\xi}_i$ applied via left multiplication. The pose refinement problem can then be expressed as

$$\boldsymbol{\xi}_i^* = \arg\min_{\boldsymbol{\xi}_i} \sum_{k=1}^{m_\tau} \left\| \tilde{\boldsymbol{b}}_{i,\tau,k} - \text{proj}\big(\exp(\boldsymbol{\xi_i}) \tilde{\boldsymbol{T}}_\tau^{B_i W} \boldsymbol{p}_k^W \big) \right\|,$$

which seeks the perturbation $\boldsymbol{\xi}_i$ that minimizes reprojection error over the m_τ visible features. This formulation provides a local approximation of the VIO update step, capturing the information contribution of 2D feature tracks to pose estimation.

Following [14], [15], the Fisher information matrix for agent i at sampled pose τ is

$$\Lambda_{\text{prior},i,\tau} := \Sigma_{\text{p},i,\tau}^{-1} = J_{\boldsymbol{\xi}_i,\tau}^\top \Sigma_{\boldsymbol{b},i,\tau}^{-1} J_{\boldsymbol{\xi}_i,\tau}, \tag{9}$$

where $J_{\boldsymbol{\xi}_i,\tau}$ is the Jacobian of the reprojection residuals with respect to the perturbation $\boldsymbol{\xi}_i$, and $\Sigma_{\boldsymbol{b},i,\tau}$ is the covariance of the 2D feature measurements visible at pose τ. For each feature k, the Jacobian is defined as

$$J_{\boldsymbol{\xi}_i,\tau,k} = \frac{\partial \boldsymbol{b}_{i,\tau,k}}{\partial \boldsymbol{\xi}_i},$$

where $\boldsymbol{p}_{i,\tau,k}^{B_i}$ is the feature position in the body frame B_i and $\boldsymbol{b}_{i,\tau,k}$ its image projection. The measurement covariance $\Sigma_{\boldsymbol{b},i,\tau}$ accounts for both anisotropic motion blur along the feature's image-plane velocity and isotropic sensor noise. The detailed construction of $\Sigma_{\boldsymbol{b},i,\tau}$ and $J_{\boldsymbol{\xi}_{i,\tau}}$ follows [15], where feature velocities are derived from the candidate trajectory velocities $\boldsymbol{v}_{i,c}(t)$. The resulting $\Sigma_{\boldsymbol{b},i,\tau}$ is block-diagonal over all $m_{i,\tau}$ features observed at pose τ.

Expression (9) therefore approximates the information associated with the prior term in (4) at a sampled future pose, quantifying the expected contribution of independent VIO to the evolution of the prior uncertainty $\Sigma_{\boldsymbol{\xi}_i}$.

For inter-agent perception, we account for the information gain from the measurement terms in (4), which arise from pairwise covisibility of 3D features. The information contribution of a shared feature k observed by agent i at sampled pose τ is

$$\Lambda_{\text{meas},i,\tau,k} = J_{\xi_i,k}^{p}{}^\top \Sigma_{\text{m},i}^{-1} J_{\xi_i,k}^{p},$$

where $J_{\xi_i,k}^p$ is the Jacobian of the feature residual with respect to $\boldsymbol{\xi}_i$ derived in (8), and $\Sigma_{\text{m},i}$ is the covariance of the 3D feature measurement in frame B_i. Summing over all shared features visible at pose τ, and assuming independent measurement noise across features gives

$$\Lambda_{\text{meas},i,\tau} = \sum_{k=1}^{m_{ij}} \Lambda_{\text{meas},i,\tau,k}. \tag{10}$$

Since we are interested in decentralized planning for agent i, the total information matrix at pose τ is constructed from the prior and measurement terms derived in (9) and (10):

$$\Lambda_{i,\tau} = \Lambda_{\text{prior},i,\tau} + \Lambda_{\text{meas},i,\tau}. \tag{11}$$

Finally, we quantify the perception-based reward using the log-determinant of the information matrix, which provides a measure of the information content of the trajectory by penalizing large uncertainty volumes

$$R_{\text{perc},i} = \frac{1}{n_{\text{poses}}} \sum_{\tau=1}^{n_{\text{poses}}} \log\det(\Lambda_{i,\tau}), \tag{12}$$

where the information volume is quantified and normalized by n_{poses} to ensure fair comparison between trajectories of different lengths, thereby encouraging planning toward trajectories that improve localization performance.

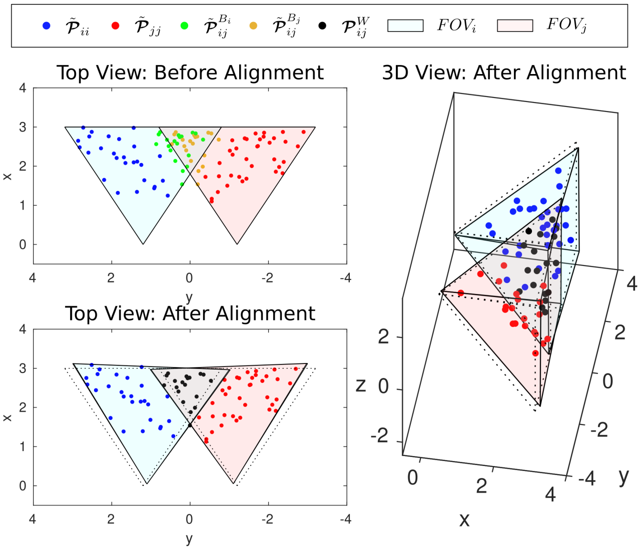


Fig. 3: Simple alignment example. The proposed algorithm corrects the poses of two neighboring vehicles using shared feature information. A systematic offset is injected into the feature positions for clarity in illustrating the effect of alignment.

V. Results

In this section, we evaluate the proposed multi-agent planning framework using controlled simulations and dataset example. Results include qualitative illustrations and quantitative comparisons against baseline methods.

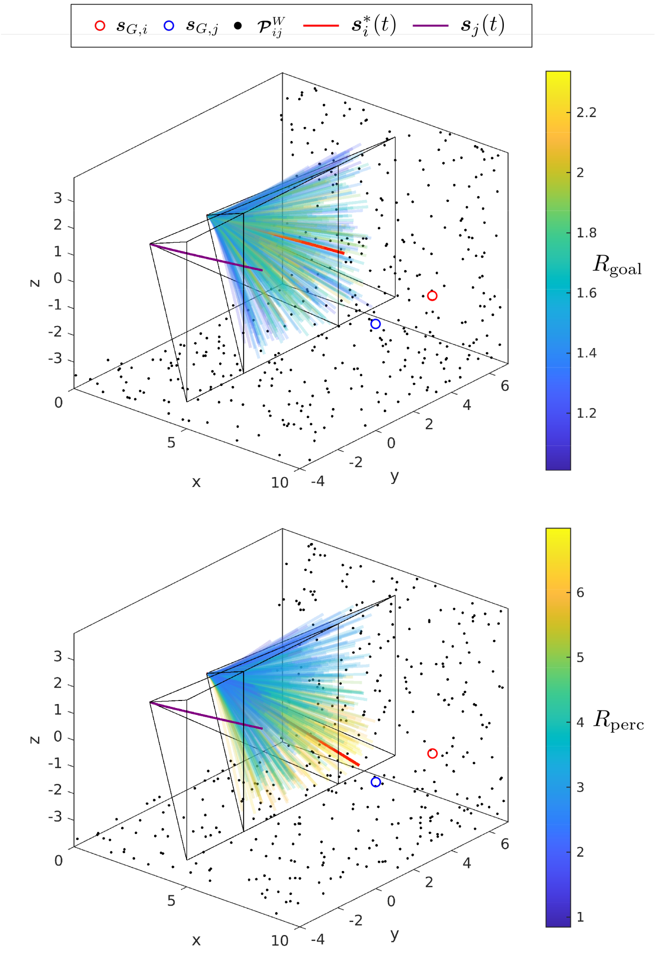


Fig. 4: Visualization of individual reward terms in trajectory selection. The optimal trajectory selected by the goal progress reward moves directly toward the goal, while the trajectory selected by the perception reward tries to maximize feature visibility.

A. Frame alignment example

We first demonstrate the frame alignment procedure with a simple two-agent example. The purpose of alignment is to anchor the relative pose between agents, leading to a collective and coupled vision drift. This mitigates unbounded divergence of the relative formation and allows a unified planning map across agents. To illustrate this, we inject systematic errors into the shared features, as shown in Fig. 3. Before alignment, the features observed by the left agent are biased to the left in its believed world frame, while those observed by the right agent are biased to the right. After solving the alignment problem, the transformations between the individual agent frames and the consensus world frame are estimated, resulting in a consistent fused map. The top-down and 3D views in Fig. 3 show how landmarks from both agents are brought into alignment. The final union of the landmarks (blue, red, and black points) forms the set $\mathcal{P}^W$, which can be subsequently used for multi-agent planning.

B. Planning reward contribution

We qualitatively illustrate how the proposed reward terms influence trajectory selection. For this example, we examine the single-step planning problem of an agent at a given instance in an environment with a floor and a wall containing features, as shown in Fig. 4. The known planned trajectory of its neighbor is shown in purple. Candidate trajectories are generated using the minimum-jerk method of [19], and the optimal one is selected following Algorithm 1. Fig. 4 shows the perception and progress rewards for 500 candidates, whose weighted sum defines the objective in (1). The perception-aware term steers the vehicle toward feature-rich regions, while the progress term favors faster goal advancement. With appropriate weighting, the selected trajectory achieves both efficient goal progress and improved estimator robustness.

C. Shared perception example

We illustrate the benefit of sharing feature information through a simple example. Fig. 5 shows two agents in an environment where features are unevenly distributed, with a large region on the left containing no features. Such texture-sparse areas often cause state estimation to degrade or fail. At each planning step, agents are restricted to using only the features that lie within their own field of view or that of a communicating neighbor. When the agents share their observations, the left agent can leverage the features visible to the right agent and plan a trajectory toward a feature-rich region. In contrast, when using only its own map (with the same reward function and initial conditions), the left agent enters the feature-sparse region and remains there without observations, since it lacks awareness of the features on the right. In practice, following this red trajectory would likely cause the VIO to fail due to the absence of trackable features. This example highlights how information sharing helps agents avoid poorly observable areas and maintain robust state estimation.

D. Frame alignment with flight data

We evaluate the proposed alignment module on a custom dataset collected from two UAVs performing a controlled hover experiment. Fig. 6 shows the dataset prior to error injection, with VIO trajectories and tracked feature points. The vehicles maintain a nominal 0.6 m separation along the x-axis, each equipped with an Intel RealSense D455 camera and running OpenVINS independently on the stereo feed. Motion-capture data is recorded as ground truth. At every step, the agents exchange their local feature maps, which are aligned to form a consistent shared representation. This setup allows us to assess how alignment corrects relative pose errors and recovers inter-agent geometry. Table II reports the RMSE of the inter-agent separation, averaged over the 30-second stationary hover phase of the dataset, after injecting a 0.5 m translational drift along a single axis, with and without alignment. Across all three directions, alignment reduces the error by about 60 %, demonstrating its effectiveness in mitigating relative drift.

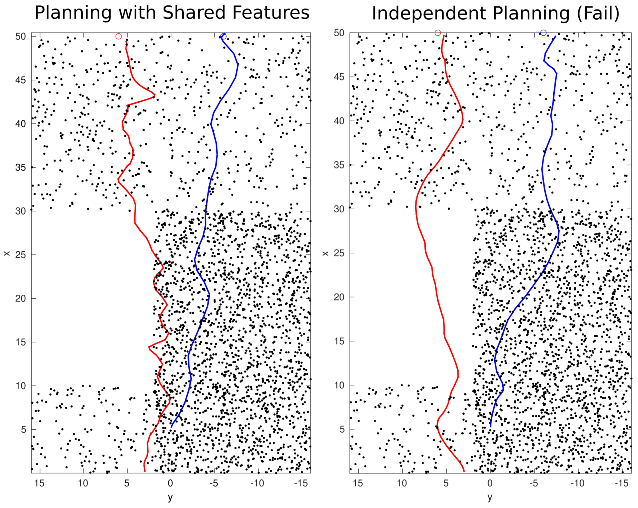


Fig. 5: Comparison of planning with and without shared feature information (top view). Blue and red lines show agent trajectories. With information sharing (left), the red agent avoids the featureless region by using its neighbor's observations, whereas without sharing (right) it enters the feature-less area and fails.

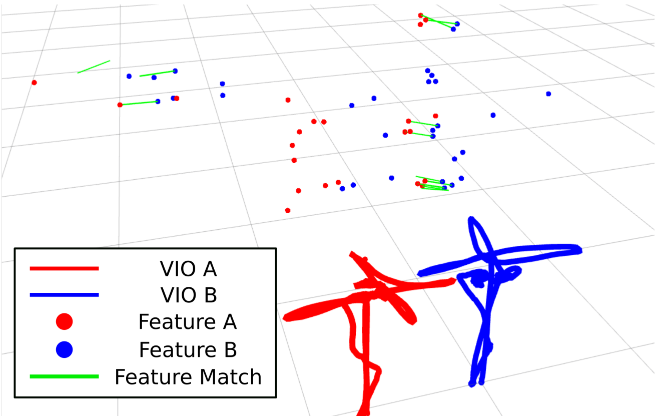


Fig. 6: Illustration of the UAV motion paths from our dataset. Two UAVs were initialized with a nominal 0.6 m separation along the x-axis, and manually moved into place before beginning a stationary hover. During this period, the agents independently run OpenVINS and exchange feature maps at each timestep for alignment.

TABLE II: RMSE of inter-agent separation distance $d(t) = \|\mathbf{s}_1(t) - \mathbf{s}_2(t)\|$ before and after correction, under $0.5\,\mathrm{m}$ injected translational drift.

Drift Direction	RMSE Old (m)	RMSE New (m)	% Reduction
X	0.510	**0.212**	58.3
Y	0.165	**0.058**	65.0
Z	0.219	**0.082**	62.7

TABLE III: Results over 100 runs (mean values). "Visible" and "Covisible" are feature counts; "Error A/B" are final position errors of the two agents.

Case	Method	# Visible	# Covisible	Error A	Error B
1	perception-agnostic	149	18	**0.52**	0.56
	perception-aware	**169**	**30**	0.80	0.56
2	perception-agnostic	132	0	**0.23**	**0.24**
	perception-aware	**141**	0	0.39	0.50
3	perception-agnostic	43	3	1.07	1.06
	perception-aware	**72**	**8**	**0.96**	**1.01**

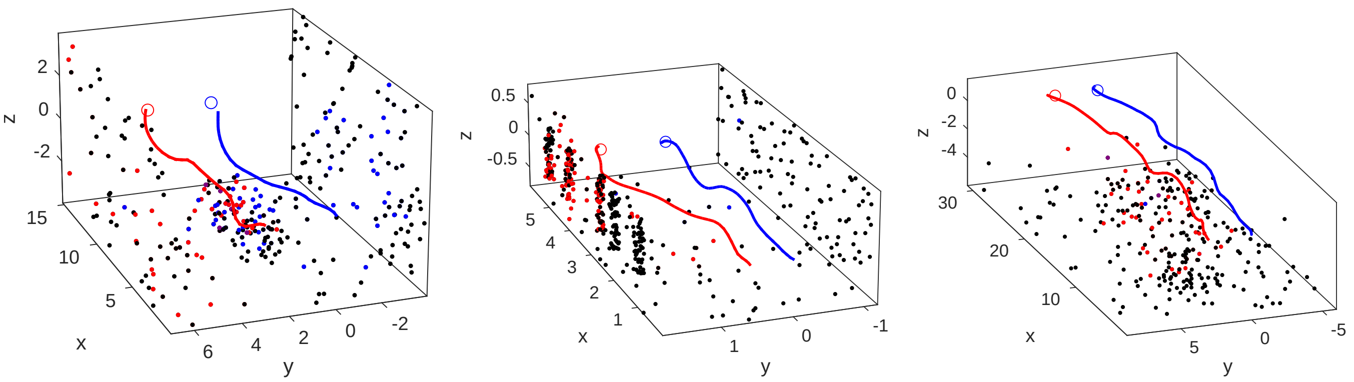
E. Multi-agent receding-horizon planning evaluation

Finally, we evaluate the proposed framework in a receding-horizon setting, where each agent replans periodically and communicates its planned horizon to neighbors. Fig. 7 illustrates results in three representative environments designed to mimic realistic deployment scenarios for multi-UAV systems. Visible features and covisible features between agents are highlighted to show the perception-aware behavior of the planner. In each experiment, trajectories for both agents are planned using the proposed framework. We generate $n_{\text{traj}} = 100$ candidate trajectories per planning step, each sampled at $n_{\text{poses}} = 10$ intermediate poses. The trajectory endpoints are sampled within the agent's field of view, following the procedure of Fig. 4, with durations drawn from $T \in [1, 3]$ s. Once the optimal trajectory is selected, its polynomial representation is shared with the other agent, and the trajectory is executed for a short horizon of $t = 0.1$ s before replanning.

Table III summarizes the quantitative outcomes, with each trajectory simulation repeated 100 times. The number of visible features is used as a proxy for estimator robustness, as each adds an independent constraint to the Fisher information matrix. While spatial distribution also matters, feature count correlates strongly with estimation accuracy. Additionally, since features are generated uniformly in our simulations, this metric remains reliable without pathological clustering cases. Across all three scenarios, the proposed approach significantly increases both visible and covisible features while maintaining comparable goal-reaching performance. Across all three cases, perception-aware planning increases the number of visible and covisible features, improving the quality of observations. However, this comes with a modest increase in final position error in Case 2 and for one agent in Case 1. In Case 3 and for the other agent in Case 1, goal accuracy is maintained or improved. Overall, these results show that the perception-aware reward consistently enhances feature observability and estimator robustness, while only trading off goal accuracy in some scenarios.

VI. Conclusion

We presented a decentralized planning framework that incorporates perception quality into trajectory optimization for multi-UAV systems. The framework aligns VIO outputs across agents to build a consistent shared map, which is then used to plan perception-aware trajectories in a receding-horizon fashion. Simulations show that our approach in-



(a) Case 1

(b) Case 2

(c) Case 3

Fig. 7: Receding-horizon planning results across three environments: (a) room with a central obstacle, (b) hallway with structured features, (c) unstructured outdoor terrain. Red and blue lines show the agents' trajectories; red and blue points are features observed uniquely by each agent; purple points are shared features observed by both agents simultaneously.

creases visible and covisible features, improves coordination, and maintains goal-reaching performance compared to perception-agnostic baselines. Future work includes experimental validation, ablation studies, modeling anisotropic landmark uncertainties from the camera projection model, incorporating formation objectives into the planning cost, and scaling to larger teams to further improve coordination and robustness.

ACKNOWLEDGMENT

This work was supported by the Agriculture and Food Research Initiative (AFRI) Competitive Grant no. 2020-67021-32855/project accession no. 1024262 from the USDA National Institute of Food and Agriculture.

REFERENCES

[1] Y. Zhou, B. Rao, and W. Wang, "Uav swarm intelligence: Recent advances and future trends," *Ieee Access*, vol. 8, pp. 183 856–183 878, 2020.

[2] M. Y. Arafat, M. M. Alam, and S. Moh, "Vision-based navigation techniques for unmanned aerial vehicles: Review and challenges," *Drones*, vol. 7, no. 2, p. 89, 2023.

[3] A. Ait Saadi, A. Soukane, Y. Meraihi, A. Benmessaoud Gabis, S. Mirjalili, and A. Ramdane-Cherif, "Uav path planning using optimization approaches: A survey," *Archives of Computational Methods in Engineering*, vol. 29, no. 6, pp. 4233–4284, 2022.

[4] X. Zhou, J. Zhu, H. Zhou, C. Xu, and F. Gao, "Ego-swarm: A fully autonomous and decentralized quadrotor swarm system in cluttered environments," in *2021 IEEE international conference on robotics and automation (ICRA)*. IEEE, 2021, pp. 4101–4107.

[5] J. Tordesillas and J. P. How, "Mader: Trajectory planner in multiagent and dynamic environments," *IEEE Transactions on Robotics*, vol. 38, no. 1, pp. 463–476, 2021.

[6] H. Wang, S. Zhang, Y. Sun, Z. Wang, J. Sun, and B. Zhu, "Swift: A distributed one-stage planner for efficient multi-quadrotor trajectory optimization," *IEEE Transactions on Automation Science and Engineering*, 2025.

[7] P. Geneva, K. Eckenhoff, W. Lee, Y. Yang, and G. Huang, "Openvins: A research platform for visual-inertial estimation," in *2020 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2020, pp. 4666–4672.

[8] A. I. Mourikis and S. I. Roumeliotis, "A multi-state constraint kalman filter for vision-aided inertial navigation," in *Proceedings 2007 IEEE international conference on robotics and automation*. IEEE, 2007, pp. 3565–3572.

[9] S. Leutenegger, S. Lynen, M. Bosse, R. Siegwart, and P. Furgale, "Keyframe-based visual–inertial odometry using nonlinear optimization," *The International Journal of Robotics Research*, vol. 34, no. 3, pp. 314–334, 2015.

[10] T. Qin, P. Li, and S. Shen, "Vins-mono: A robust and versatile monocular visual-inertial state estimator," *IEEE transactions on robotics*, vol. 34, no. 4, pp. 1004–1020, 2018.

[11] R. Mur-Artal, J. M. M. Montiel, and J. D. Tardos, "Orb-slam: A versatile and accurate monocular slam system," *IEEE transactions on robotics*, vol. 31, no. 5, pp. 1147–1163, 2015.

[12] P. Zhu, Y. Yang, W. Ren, and G. Huang, "Cooperative visual-inertial odometry," in *2021 ieee international conference on robotics and automation (icra)*. IEEE, 2021, pp. 13 135–13 141.

[13] T. Zhang, L. Zhang, Y. Chen, and Y. Zhou, "Cvids: A collaborative localization and dense mapping framework for multi-agent based visual-inertial slam," *IEEE transactions on image processing*, vol. 31, pp. 6562–6576, 2022.

[14] Z. Zhang and D. Scaramuzza, "Perception-aware receding horizon navigation for mavs," in *2018 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2018, pp. 2534–2541.

[15] X. Wu, S. Chen, K. Sreenath, and M. W. Mueller, "Perception-aware receding horizon trajectory planning for multicopters with visual-inertial odometry," *IEEE Access*, vol. 10, pp. 87 911–87 922, 2022.

[16] J. Lim, N. Lawrance, F. Achermann, T. Stastny, R. Girod, and R. Siegwart, "Fisher information based active planning for aerial photogrammetry," in *2023 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2023, pp. 1249–1255.

[17] K. Kondo, C. T. Tewari, M. B. Peterson, A. Thomas, J. Kinnari, A. Tagliabue, and J. P. How, "Puma: Fully decentralized uncertainty-aware multiagent trajectory planner with real-time image segmentation-based frame alignment," in *2024 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2024, pp. 13 961–13 967.

[18] D. M. Rosen, K. Khosoussi, C. Holmes, G. Dissanayake, T. Barfoot, and L. Carlone, *Certifiably Optimal Solvers and Theoretical Properties of SLAM*. Cambridge University Press, 2026.

[19] M. W. Mueller, M. Hehn, and R. D'Andrea, "A computationally efficient motion primitive for quadrocopter trajectory generation," *IEEE transactions on robotics*, vol. 31, no. 6, pp. 1294–1310, 2015.

[20] N. Bucki, J. Lee, and M. W. Mueller, "Rectangular pyramid partitioning using integrated depth sensors (rappids): A fast planner for multicopter navigation," *IEEE Robotics and Automation Letters*, vol. 5, no. 3, pp. 4626–4633, 2020.

[21] S. M. Kay, *Fundamentals of statistical signal processing: estimation theory*. Prentice-Hall, Inc., 1993.