# Exact Decentralized Receding Horizon Planning for Multiple Aerial Vehicles

Indrajeet Yadav and Herbert G. Tanner

***Abstract*— This article presents a decentralized approach to motion planning for a group of aerial vehicles that combines deliberate and reactive planning strategies to ensure multi-vehicle interception of a moving target and coordinated tracking while keeping a desired formation. A new on-board receding horizon planning and control architecture realizes each vehicle's motion plan by generating and tracking dynamically feasible trajectories, while also incorporating a reactive collision avoidance feature that allow them to avoid collisions with each other and with a priori unknown obstacles. The ability of the architecture to safely steer the vehicles to their intercepting formation time-varying positions is demonstrated in realistic Gazebo simulations.**

## I. Introduction

Transit of illicit nuclear material can potentially be detected by networks of micro aerial vehicles (MAV) equipped with lightweight commercial off the shelf (COTS) radiation detectors. In cases where the emitted radioactivity is weak enough to blend with naturally occuring background radiation, the aerial sensor platforms will need to come as close as possible to the suspected mobile target and stay with it for as long as needed for a confident decision to be reached about the latter's nature. (Note that the radiation *detection* problem is different from a radiation *mapping* problem, e.g. [6]) Such a hypothetical mission will require the MAVs to autonomously navigate in some potentially partially known cluttered environment, intercept their target, and fly in formation in proximity to it. This paper's planning and control strategy aims to achieve precisely that.

The binary hypothesis decision-making problem (i.e. is the target radioactive or not) is treated elsewhere [13], [23], and optimal (from a detection perspective) aerial vehicles formations have been identified [21]. This paper is concerned primarily with the question of how the MAVs will provably and safely achieve such formations, even when the environment in which they operate is partially known. An example of the partial environment knowledge contemplated here is a case where the vehicles may be equipped with a map of an urban area that identifies buildings and other large man-made structures, but does not necessarily includes, trees, crane structures, vehicles or pedestrians.

Existing approaches to MAV formation control in cluttered environments involve several different techniques, including vision-based control [12], coordinated multi-MAV formation trajectory planning and control [18], as well as distributed formation control with obstacle avoidance [1], [2]. In the latter approaches, the MAVs navigate their cluttered workspace by collectively identifying the largest obstacle-free convex region, and then moving while keeping the convex-hull of their positions within the identified safe region. The final MAV configuration is obtained by minimizing the error with respect to a desired formation. The key difference of the problem setup in this paper compared to the aforementioned work is that here the desired MAV formation needs to move and track a designated target.

When one approaches the problem from a MAV target tracking viewpoint, they encounter work in literature that examines the case where a *single* MAV is tracking a specific target [5], [17]. As a result, there is little consideration of how different MAV may interact with each other. Recent reports in literature where MAV formations are tasked to track a moving target are scarce (cf. [19]), unless one traces early work that involves formation control based on virtual structures [4], [8]. Still, very little is known about how groups of MAV can achieve dynamic formations when operating in constrained and partially known environments.

Feedback-based provably convergent MAV navigation in cluttered environments is challenging. Setting aside combinatorial or sampling techniques who have demonstrated results [11] but cannot enjoy the robustness of feedback, recent extension of navigation function techniques [15] that incorporate elements of reactive behavior and on-line adaptation based on sensor data have shown promise [3], [20], but have not yet been demonstrated to be effective for dynamic and high-speed MAV flight; in addition, convergence to a moving destination has not been formally established in this framework.

In contrast, there has been work that formally extends the navigation function formulation explicitly along the direction of moving destinations, and has been experimentally validated with aerial (albeit relatively slow) flight [16] (cf. [9]).

This paper builds on the body of knowledge of navigation functions with moving destinations, and integrates them within a comprehensive MAV flight planning and control architecture which utilizes receding horizon control and incorporates the full aerial vehicle dynamics, and includes a reactive, sensor-based collision avoidance component. The key contribution of the work is to develop a single motion planning strategy that can be used by a small team of aerial vehicles to track a moving target in a formation while avoiding the obstacles encountered during the interception or the tracking phase.

The planning and control approach starts from an initial condition where the desired dynamic formation around a given target has been determined and the position of each vehicle in this formation has been identified [21] and extends

Yadav and Tanner are with the Department of Mechanical Engineering at the University of Delaware {indragt,btanner}@udel.edu

This work has been supported in part by NSF under award # 1548149, and in part by DTRA under grant #HDTRA1-16-1-0039.

the state-of-the-art on the problem of cooperative dynamic target interception and tracking in cluttered environments in the following directions: (i) enables the MAVs to navigate in a cluttered and partially known environments and reactively avoid each other and unmapped objects in their workspace, (ii) offers a formal proof of convergence for the navigation function-based reference paths that steer the MAVs to their time-varying desired positions, (iii) allows generation algorithm to build on these reference paths to produce dynamically feasible reference trajectories that minimize MAV jerk, and finally (iv) integrates all these components in a new comprehensive MAV receding horizon flight planning and control architecture that modestly extends a differential geometric MAV controller [7] to offer an end-to-end solution to the problem of cooperative autonomous interception and tracking of moving targets.

Section II and Section III set the motion planning and control problem in a mathematical basis and give the architecture of the proposed approach. Section IV-A introduces the navigation function-based motion planning methodology that enables each MAV to negotiate known environmental features and chart a path to its desired configuration to the formation. This is followed by the receding horizon based dynamically feasible trajectory generation approach presented in Section IV-B and IV-C. The simulations results and conclusions have been presented in Section V and VI respectively.

## II. Problem Statement

Consider $N$ identical MAVs in the form of quadrotors which move in a shared constrained physical *workspace* $\mathcal{W}$, and assume that there is a fixed *inertial frame* defined in this space. The shared workspace is practically 2.5D, and is populated by fixed cylindrical obstacles, some of which are known a priori, and some are unknown. The portion of the workspace which does not overlap with obstacles (known or unknown) is denoted $\mathcal{F}$ and is referred to as the *free workspace*.

Denote the mass of MAV $i$ $m_i$, and its moment of inertia about the principal axes passing through its center of mass, $J_i$. (The aforementioned principal frame will henceforth be referred to as the *body-fixed frame*.) The propeller-generated thrust (magnitude) and moment about the body-fixed axes of MAV $i$ are denoted $f_i$ and $M_i$, respectively. The MAV orientation relative to the inertial frame is expressed in the form of the rotation matrix $R_i \in \mathsf{SO}(3)$, and its Cartesian position with respect to the inertial frame is represented by a vector $x_i \in \mathbb{R}^3$. The angular velocity vector expressed in the body-fixed frame of the MAV is denoted $\Omega_i$.

If $\widehat{\ }: \mathbb{R}^3 \to \mathfrak{so}(3)$ denotes the (hat) operation associating 3D vectors to elements of the 3D Lie algebra, $g$ the acceleration of gravity vector, and $e_3$ the unit vector along the 3rd local coordinate axis, then the dynamics of MAV $i$ are expressed, for $i = 1, \ldots, N$, as

$$m_i \ddot{x}_i = -m_i g e_3 + f_i R_i e_3 \tag{1a}$$

$$\dot{R}_i = R_i \hat{\Omega}_i \tag{1b}$$

$$M_i = J_i \dot{\Omega}_i + \Omega_i \times J_i \Omega_i \tag{1c}$$

The trajectory generated for each MAV should be dynamically feasible i.e., should be consistent with the higher order dynamics of the quadrotor, should enable the generation of smooth control inputs, and yield commanded thrust and angular velocities within the permissible limits. Let $f_{\min}$, $f_{\max}$ and $\Omega_{\max}$ be the limits on thrust and angular speed; then the dynamical feasibility requirement is cast as the condition that for every $i \in \{1, \ldots, N\}$

$$f_{\min} \leq f_i \leq f_{\max} \qquad \|\Omega_i\| \leq \Omega_{\max} \tag{2}$$

The goal of the MAV group is to navigate through the workspace in a collision-free manner and fall into a (moving) formation where each MAV $i$ has a designated *desired goal* $g_i(t) \in \mathbb{R}^3$. It is assumed that both $x_i(t)$ and $g_i(t)$, for all $i \in \{1, \ldots, N\}$, remain in the interior of $\mathcal{F}$ for all $t \geq 0$.

## III. System Architecture

The structure of the overall architecture for planning, control and formation generation is shown in Fig. 1. The arrows indicate the direction of information flow. Different colored blocks (blue, red and green) represent the software stack running independently on each MAV. Each MAV is supplied with the available information about the workspace topology and its time-varying destination on the formation.

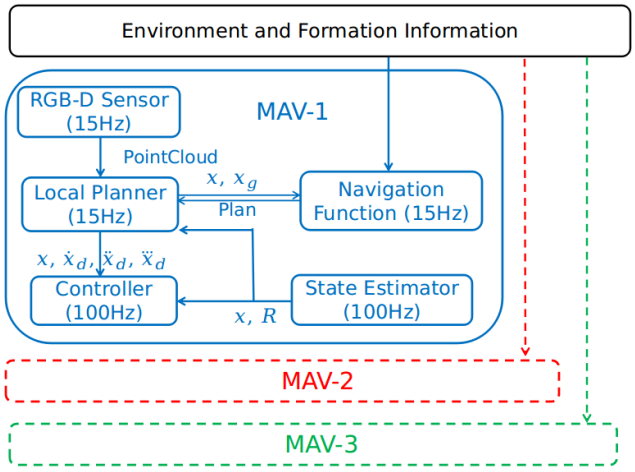



Fig. 1. System block diagram. Each (identical) colored block represents the software stack architecture on an individual MAV.

The computation steps (identical to each MAV) in the cascaded planning architecture depicted in the first two blocks of Fig. 1, reading from top to bottom, are as follows. (i) A priori information available about the workspace is utilized off-line to construct a navigation function for the particular MAV; (ii) point-cloud data from the MAV's RGBD sensor is harnessed by its reactive local planner, which operates iteratively in a receding horizon fashion at 15Hz, and shares the MAV's current (estimated) location and the MAV's desired destination with the navigation function; (iii) the navigation function receives the MAV's current and desired configuration and produces an initial *segment* of a continuous path connecting the current MAV configuration

to its moving goal (assuming no unknown obstacles are along that path); (iv) the receding horizon local planner receives the path segment from the navigation function and utilizing the most current point-cloud information generates a dynamically feasible minimum-jerk trajectory that fits this path as tightly as possible while avoiding reactively any (previously unknown) obstacles detected by the RGBD sensor; and finally, (v) a small *initial component* of that minimum-jerk reference trajectory is faithfully tracked by a provably stable differential-geometric tracking controller operating at 100Hz, which starts steering the quadrotor along the trajectory segment within the FOV.

In a typical receding horizon fashion, before the end of first trajectory segment is reached, the navigation function planning block updates the collision-free path segment; subsequently the receding horizon planner updates the reference trajectory segment and stitches the old and the new trajectory segments smoothly through an (minimum jerk) optimization process that is detailed in Section IV-C. The re-planning and trajectory tracking process is repeated during the entire target tracking process and terminates when the MAV is the neighborhood of its moving formation destination.

## IV. MAV Motion Planning

### A. Deliberate Global Planning

The control objective of MAV $i$ is achieved whenever the function (dependence on time will be dropped for brevity, but in the remainder it is understood that $g_i$ is a function of time)

$$J_i(x_i, g_i) = \|x_i - g_i\| \tag{3}$$

converges to zero as time tends to infinity.

It is assumed that all known obstacles $j = 1, \ldots, m$ in $\mathcal{W}$ are spherical with implicit representations of the form

$$\beta_j(x_i) = \|x_i - o_j\|^2 - \rho_j^2$$

where $o_j$ and $\rho_j$ are their centers and radii, respectively, whereas the workspace boundary is captured by

$$\beta_0(x_i) = \rho_0^2 - \|x_i - o_0\|^2$$

The assumption regarding the spherical shape of obstacles is not considered particularly restricting since it is known that there exist diffeomorphisms mapping environments with more general, star-shaped obstacles, to workspaces with spherical obstacles like $\mathcal{W}$ [15].

Proximity of MAV $i$ to (known) obstacles in $\mathcal{W}$ can be captured by the scalar function $\beta(x_i) = \prod_{j=0}^{m} \beta_j(x_i)$. Then an artificial potential function can be constructed in the form

$$\varphi_i(x_i, g_i) = \frac{J_i^2(x_i, g_i)}{\left[J_i(x_i, g_i)^{2\kappa} + \beta(x_i)\right]^{1/\kappa}} \tag{4}$$

and its parameter $\kappa$ can always be appropriately tuned so that (4) can acquire navigation function properties [9], [16]. This essentially means that $g_i$ is the sole unique minimum of $\varphi_i$, and any other critical point in $\mathcal{F}$ is a saddle with an attraction region which is a set of measure zero.

Under the assumption that $g_i(t)$ and $\dot{g}_i(t)$ is known, it can be shown that a gradient descent-based feedback law with appropriate feedforward can ensure that a hypothetical single-integrator system $\dot{x}_i = u$ converges to $g_i$ from almost every free workspace initial configuration:

**Proposition 1.** *Consider the dynamics $\dot{x}_i = u_i$. Then there exists sufficiently large $k_{\varphi_i} > 0$ such that*

$$u_i = -k_{\varphi_i} \frac{\partial \varphi_i}{\partial x_i} \tag{5}$$

*results in* $\lim_{t\to\infty} x_i(t) = g_i(t)$.

*Proof:* (Sketch) Note first that $\varphi_i$ is lower bounded by 0. Its time derivative evaluates to

$$\dot{\varphi}_i(x_i, g_i) = - k_{\varphi_i} \left\| \frac{\partial \varphi_i}{\partial x_i} \right\|^2 - \frac{2\beta(x_i)\langle (x_i - g_i), \dot{g}_i \rangle}{\left[\|x_i - g_i\|^{2\kappa} + \beta(x_i)\right]^{(1/\kappa)+1}} \tag{6}$$

For bounded target speed $\dot{g}_i$ within $\mathcal{F}$ and outside the neighborhood $S$ of critical points of $\varphi_i$, one can pick

$$k_{\varphi_i} > \max_{x_i \in \mathcal{F} \setminus S} \frac{2\beta(x_i)\langle (x_i - g_i), \dot{g}_i \rangle}{\left[\|x_i - g_i\|^{2\kappa} + \beta(x_i)\right]^{(1/\kappa)+1} \cdot \|\nabla \varphi_i\|^2} < \infty$$

so that the squared norm term in (6) dominates, establishing sign definiteness for $\dot{\varphi}_i$. Now the right-hand-side of (6) is bounded and uniformly continuous everywhere in $\mathcal{F}$, while $\varphi_i(x_i, g_i) \in [0, 1]$. As a result, $\dot{\varphi}_i \to 0$ as $t \to \infty$, which implies that $x_i \to g_i$. □

Note also that since $-\frac{\partial \varphi_i}{\partial x_i}$ always points away from (known) obstacle surfaces, the paths generated by (5) will never intersect with the boundaries of the known work space.

Given Proposition 1, now one knows that properly moderated gradient decent paths generated by $\varphi_i$ through (5) are guaranteed to connect the current position of MAV $i$ to its target $g_i$ while avoiding (known) obstacles.

A path can now be generated from the current MAV location to the goal by numerical integrating point mass dynamics with the control input defined by (5). A small initial *segment* of this path is utilized by the reactive planner introduced in next section.

### B. Reactive Local Planning

Given a path segment produced by the navigation function, and point-cloud input data from the onboard RGBD sensor, the receding horizon planner's goal is to continuously generate feasible, safe, and locally optimal trajectories. Tracking those trajectories, the MAV can deliberatively steer around known obstacles, and reactively avoid previously unknown obstacles within its FOV while making progress toward the dynamic navigation goal [22]. This section presents a brief description of the approach followed.

If the components of the position vector $x_i$ of MAV $i$ are denoted $(p_{xi}, p_{yi}, p_{zi})$, and $\psi_i$ denotes its yaw angle, then a continuously differentiable trajectory in the $(x_i, \psi_i)$ space is ensured to be compatible with the MAV's dynamics, which are known to be differentially flat [10]. If, in addition, it so

happens that such a trajectory respects the input feasibility constraints (2), then the MAV state trajectory that is associated with the differentially flat trajectory is guaranteed to be dynamically feasible.

The field of view (FOV) is assumed to be contained in a right rectangular pyramid, having its apex at the base frame of the RGBD sensor, with its depth direction aligned with the local $x$ MAV axis, and its height of determined by the sensor's range. For computation purposes, the volume of the FOV pyramid is discretized and reduced to a grid of fixed resolution. Then consider a ray cast from the apex to each of the grid points inside the FOV pyramid volume. Such rays represent potential local directions of motion for the MAV. Any ray in proximity to an RGBD point-cloud element is discarded as non collision-free. Separating the collision-free rays can be done efficiently using a KD-tree data structure [22]. Figure 2 shows this process. A path segment (indicated as magenta line) is received by the navigation function-based planner of Section IV-A. The end point of this segment is referred to as *intermediate point*.

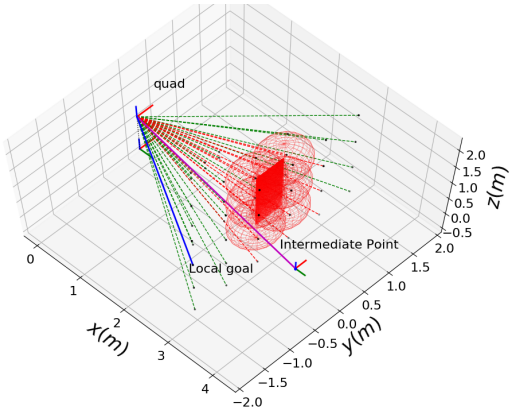



Fig. 2. Checking for collisions. Triad mark the MAV's current COG and the intermediate point, which is the end point of the path segment generated by navigation function (indicated by magenta line). Rays in the FOV intersecting with obstacles reflected in point cloud data (red) are discarded. Green rays represent collision-free motion directions and blue ray is the optimal ray along with reference trajectory is generated.

Which of all the FOV rays will eventually be selected as the preferred local direction of motion is determined by a reactive optimization heuristic that strikes a balance between three desirable factors: (a) collision avoidance, according to which rays are penalized relative to how close they would bring the MAV to an obstacle detected within the FOV; (b) convergence, according to which rays with end points further away from the intermediate point are penalized; and (c) inter-MAV coordination, based on which —and in lieu of any communication between the aerial vehicles— counter-clockwise maneuvers for avoidance of collision to unknown obstacles detected within the MAV's FOV are preferable. The latter factor (referred to as the directional cost) aims at resolving situations where two MAV cross paths on their way to their formation destinations. The counter-clockwise rule —the actual direction is arbitrary and immaterial; it could be equivalently set as a clockwise rule— is designed to increase the distance between a pair of MAVs in a collision course.

In the face of the practical limitations stemming from (a) the uncertainty regarding the workspace topology, (b) the very limited FOV of the MAVs, (c) lack of centralized coordination and communication between the MAVs, and (d) the need to reactively avoid both static (obstacles) and moving (other MAVs) objects, it is extremely challenging to fully recover the guarantees of collision avoidance provided by Proposition 1. Nonetheless, as long as the unknown obstacles are sparse and of size comparable to that of the MAVs, the occasional local reactive action becomes an intermittent perturbation which does not disturb convergence.

Denote $p$ the total number of collision-free rays (assumed for mere presentation simplicity to be the same for MAVs), and $d_k$ the Euclidean distance between the end point of the $k^{\text{th}}$ ray and the intermediate point, for $k = 0, \ldots, p$. This $d_k$ represents the *convergence cost* for the ray $k$.

Set $d_{\max} \triangleq \max_k d_k$, and let $\hat{r} \geq r$ be an additional safety margin (uniform inflation around the detected obstacle boundaries). Letting $\rho_k$ be the minimum distance to the nearest obstacle out of all interior points on ray $k$, the *collision cost* for ray $k$ is calculated as [22]

$$c_{\text{coll}_k} = \begin{cases} \frac{1+\hat{r}^4}{\hat{r}^4} \cdot \frac{[(\rho_k - r)^2 - \hat{r}^2]^2}{1+[(\rho_k - r)^2 - \hat{r}^2]^2} & \text{if } \rho_k - r \leq \hat{r} \\ 0 & \text{otherwise} \end{cases}$$

While the collision and convergence costs are calculated in the standard MAV body frame, the third factor i.e., the directional cost is calculated in the (RGBD) sensor's frame. This frame has $z$ and $y$ axes aligned to the depth and vertical sensor directions, respectively, while its $x$ axis distinguishes clockwise from counter-clockwise turns within the sensor's FOV. Let $x_{\min}$ and $x_{\max}$ be the minimum and maximum local end point $x$ coordinates out of all non-colliding rays. Then the *directional cost* of ray $k$ is taken to be

$$c_{\text{dir}_k} = \frac{x_k - x_{\min}}{x_{\max} - x_{\min}}$$

Now for positive scalar weights $k_1$, $k_2$, and $k_3$, all in the $(0, 1)$ range, the *total cost* of ray $k \in [0, \ldots, p]$ is

$$c_k = k_1 \tfrac{d_k}{d_{\max}} + k_2\, c_{\text{coll}_k} + k_3\, c_{\text{dir}_k}$$

The end point of the ray that has the lowest total cost is called the *local goal* (because it is within the FOV) of the reactive local planner (Fig. 2).

### C. Receding Horizon Trajectory Tracking

With the local goal in the FOV selected, the problem now reduces to generating a dynamically feasible and locally optimal reference trajectory from the MAV current 3D position to the local goal, tracing the optimal ray. In the spirit of receding horizon control, this reference trajectory has a predefined *planning horizon* $T$, and boundary conditions (in terms of MAV $i$ position, velocity, acceleration, and jerk) denoted $\mathsf{x}_{i0} = \left[x_{i0}^\mathsf{T}, \dot{x}_{i0}^\mathsf{T}, \ddot{x}_{i0}^\mathsf{T}, \dddot{x}_{i0}^\mathsf{T}\right]^\mathsf{T}$ and $\mathsf{x}_{iT} = \left[x_{iT}^\mathsf{T}, \dot{x}_{iT}^\mathsf{T}, \ddot{x}_{iT}^\mathsf{T}, \dddot{x}_{iT}^\mathsf{T}\right]^\mathsf{T}$, respectively, where $\mathsf{x}_{i0}$ is matching the current state of the MAV, $x_{iT}$ the position of the local goal in the FOV, $\dot{x}_{iT}$ the speed of the target $\|\dot{g}_i\|$ along the direction of the optimal ray while $\ddot{x}_{iT} = \dddot{x}_{iT} := 0$. The planning horizon $T$ is

determined adaptively. Assuming a hypothetical trapezoidal speed profile along the length of the optimal ray using the aforementioned boundary conditions at $\mathsf{x}_i 0$ and $\mathsf{x}_{iT}$, and imposing a constraint on maximum speed consistent with (2), one can calculate the resulting time to traverse this path and thus numerically determine a value for $T$.

Let the optimal ray be divided into $n_p$ equal segments, defining waypoints at their endpoints, and let $\Delta t_\ell$ denote the time difference between the time stamp of two successive waypoints; it follows that $T = \sum_{\ell=1}^{n_p} \Delta t_\ell$. Each segment of the reference trajectory $\mathsf{x}_\ell : [0, \Delta t_\ell] \to \mathbb{R}^{12}$, for $\ell = 1, \ldots, n_p$ and $\mathsf{x}_\ell(\Delta t_\ell) = \mathsf{x}_{\ell+1}(0)$, can now be produced as a solution to a minimum jerk optimization problem [10], [14]

$$\begin{cases} \arg\min_{\mathsf{x}_\ell} \sum_{\ell=1}^{n_p} \int_0^{\Delta t_\ell} \left\| \frac{d^3 x_\ell}{dt^3} \right\|^2 \mathrm{d}t \\ \text{subject to} \\ \frac{\mathrm{d}^n x_\ell}{\mathrm{d}t^n} \big|_{\Delta t_\ell} = \frac{\mathrm{d}^n x_{\ell+1}}{\mathrm{d}t^n} \big|_0 \quad n = 0, \ldots, 3 \\ \mathsf{x}(0) = \mathsf{x}_0, \;\; \mathsf{x}(T) = \mathsf{x}_T \end{cases} \tag{7}$$

Problem (7) is converted to a quadratic program (QP) and efficiently solved using standard solvers. The subscripts for the MAV is dropped for clarity. Since yaw is dynamically decoupled from MAV position, the trajectory reference for $\psi_i$ can be generated independently using a third order time polynomial [22]. Once the full reference trajectory is generated at a given receding horizon cycle, it is checked to ensure that the nominal thrust and angular velocity conforms to the dynamic constraints (2), and in case of constraint violation velocities are progressively scaled down to yield less aggressive but feasible MAV maneuvers. Once the final reference trajectory is obtained, an initial portion of *control horizon* $T_c \ll T$ is tracked through a variation [22] of a differential-geometric motion controller (cf. [7]), at which time the receding horizon control cycle is complete and a new motion planning process is initiated. Figure 3 illustrates an example of the outcome of the receding horizon strategy for the case shown in Fig. 2.

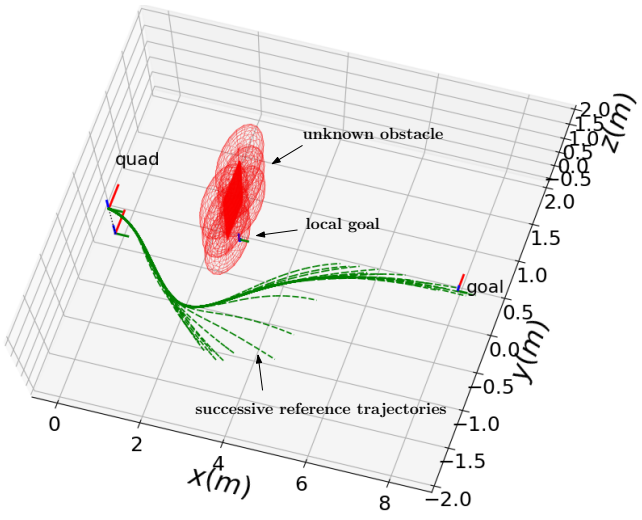



Fig. 3. Trajectory Generation. Dashed green lines show generated reference trajectories while solid green curve is the trajectory of the MAV as it tracked the trajectory segments consecutively.

## V. Results

This section presents the results of numerical simulation conducted using ROS and GAZEBO. Figure 4(a) shows a 2.5D sphere world with six known and two unknown obstacles. The target moves along a straight (black) line passing through the center of the workspace and in between the two unknown obstacles, and two MAVs, their paths marked by the magenta and green curves, are intercepting and tracking it in a formation which has them maintain a distance of 1 m in the $x$ and $y$ directions. The figure insets show the reactive collision avoidance maneuvers in more detail; the top left inset illustrates how MAV-2 modifies its trajectory to avoid MAV-1 which is already in persuit, and the bottom right inset shows how the vehicle avoids one unknown obstacle.

In Fig. 4(b) one sees the evolution of the navigation functions of the two MAVs over time. The value of the navigation function decreases and converges to zero. From the plot of Fig. 4(b) one can observe a slight initial increase in the value of the functions that corresponds to the MAVs taking off their original landing locations, and a temporary increase (see inset) for one of the MAVs as it performs a collision avoidance maneuver for the other MAV encountered. A short video depicting gazebo simulations can be found at

Figure 5 illustrates a more cluttered (in terms of unknown obstacles) scenario with three MAVs which fall into a triangle formation as they intercept their target. The inset details an instance of the MAVs reactively avoiding collisions while attempting to maintain their formation. A link to GAZEBO and RVIZ simulations can be found at `https://youtu.be/tRXes4v_bWk`.

## VI. Conclusions

Local reactive, and global deliberate motion planners for MAVs, can be effectively combined within a receding horizon framework to address a problem of completely decentralized multi-MAV tracking and interception of a moving target in a partially known environment. Based on available workspace information, the deliberate global motion planner can offer guarantees of safety and convergence; given its sensor-based, and reactive nature the local planner cannot match this. Nonetheless, a receding horizon control strategy that concurrently engages both planners in an iterative fashion is demonstrated to be effective in steering the MAVs to their desired intercepting states, and due to the feedback nature of the overall architecture (Fig. 1), the local reactive action does not inhibit the convergence of MAVs to their target.

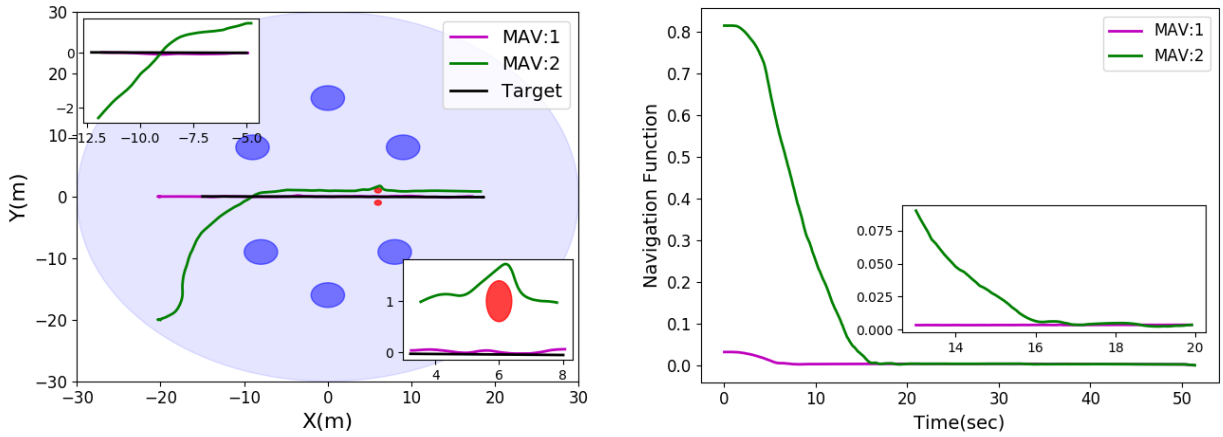
## References


[1] J. Alonso-Mora, E. Montijano, M. Schwager, and D. Rus. Distributed multi-robot formation control among obstacles: A geometric and optimization approach with consensus. In *2016 IEEE International Conference on Robotics and Automation*, pages 5356–5363, May 2016.

[2] Javier Alonso-Mora, E. Montijano, T. Nägeli, O. Hilliges, M. Schwager, and D. Rus. Distributed multi-robot formation control in dynamic environments. *Autonomous Robots*, Jul 2018.

[3] Omur Arslan and Daniel E Koditschek. Sensor-based reactive navigation in unknown convex sphere worlds. *The International Journal of Robotics Research*, 38(2-3):196–223, 2019.

[4] Yongcan Cao and Wei Ren. Distributed coordinated tracking with reduced interaction via a variable structure approach. *IEEE Transactions on Automatic Control*, 57(1):33 –48, 2012.




(a) (b)

Fig. 4. Tracking and interception in a partially known workspace with 2 MAVs. (a) Light blue shading marks the workspace area, with darker blue showing known obstacles and red marking unknown obstacles. The black curve indicates the target's path; the paths of the two MAVs are shown in magenta and green, respectively, with the insets showcasing different reactive collision avoidance maneuvers. (b) The evolution of the navigation functions over time. Inset images shows the variation of the function when the two MAVs intercept the target.

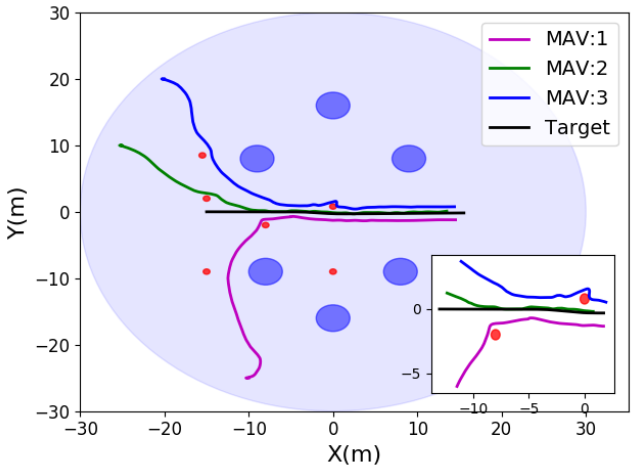



Fig. 5. An example of a more cluttered and unknown workspace used in Second Simulation with three MAVs. Different MAV paths are shown as magenta, green and blue. Dark blue areas mark known, and red ones unknown obstacles. The inset image shows the obstacle avoidance maneuver performed by the MAVs at the center of the workspace.


[5] J. Chen, T. Liu, and S. Shen. Tracking a moving target in cluttered environments using a quadrotor. In *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 446–453, Oct 2016.

[6] Z. Cook, M. Kazemeini, and A. Barzilov. Low-altitude contour mapping of radiation fields using uas swarm. *Intelligent Service Robotics*, pages 1–12, April 2019.

[7] T. Lee, M. Leoky, and N. H. McClamroch. Geometric tracking control of a quadrotor uav on se(3). In *Proceedings of 49th IEEE Conference on Decision and Control*, pages 5420–5425, 2010.

[8] M. Anthony Lewis and Kar-Han Tan. High precision formation control of mobile robots using virtual structures. *Autonomous Robots*, 4:387–403, 1997.

[9] C. Li and H. G. Tanner. Navigation functions with time-varying destination manifolds in star worlds. *IEEE Transactions on Robotics*, 35(1):35–48, Feb 2019.

[10] D. Mellinger and V. Kumar. Minimum snap trajectory generation and control for quadrotors. In *Proceedings of IEEE International Conference on Robotics and Automation*, pages 2520–2525, May 2011.

[11] Kartik Mohta et al. Fast, autonomous flight in gps-denied and cluttered environments. *Journal of Field Robotics*, 35(1):101–120, 2018.

[12] E. Montijano, E. Cristofalo, D. Zhou, M. Schwager, and C. Sagüés. Vision-based distributed formation control without an external positioning system. *IEEE Transactions on Robotics*, 32(2):339–351, April 2016.

[13] Chetan D. Pahlajani, Jianxin Sun, Ioannis Poulakakis, and Herbert G. Tanner. Error probability bounds for nuclear detection: Improving accuracy through controlled mobility. *Automatica*, 50(10):2470–2481, 2014.

[14] Charles Richter, Adam Bry, and Nicholas Roy. Polynomial trajectory planning for aggressive quadrotor flight in dense indoor environments. In Masayuki Inaba and Peter Corke, editors, *Robotics Research*, volume 114 of *Springer Tracts in Advanced Robotics*, pages 649–666. Springer International Publishing, Cham, 2016.

[15] E. Rimon and D. E. Koditschek. Exact robot navigation using artificial potential functions. *IEEE Transactions on Robotics and Automation*, 8(5):501–518, Oct 1992.

[16] Jianxin Sun and Herbert G. Tanner. Constrained decision-making for low-count radiation detection by mobile sensors. *Autonomous Robots*, 39(4):519–536, 2015.

[17] J. Thomas, J. Welde, G. Loianno, K. Daniilidis, and V. Kumar. Autonomous flight for detection, localization, and tracking of moving targets with a small quadrotor. *IEEE Robotics and Automation Letters*, 2(3):1762–1769, July 2017.

[18] Matthew Turpin, Nathan Michael, and Vijay Kumar. Trajectory design and control for aggressive formation flight with quadrotors. *Autonomous Robots*, 33(1):143–156, Aug 2012.

[19] Luis R. Valbuena and Herbert G. Tanner. Flocking, formation control and path following for a group of mobile robots. *IEEE Transactions on Control Systems Technology*, 23(4):1268–1282, 2015.

[20] V. Vasilopoulos, G. Pavlakos, S. L. Bowman, J. D. Caporale, K. Daniilidis, G. J. Pappas, and D. E. Koditschek. Reactive semantic planning in unexplored semantic environments using deep perceptual feedback. *IEEE Robotics and Automation Letters*, 5(3):4455–4462, 2020.

[21] I. Yadav and H. G. Tanner. Mobile radiation source interception by aerial robot swarms. In *Proceedings of International Symposium on Multi-Robot and Multi-Agent Systems*, pages 63–69, 2019.

[22] I. Yadav and H. G. Tanner. Reactive receding horizon planning and control for quadrotors with limited on-board sensing. In *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 7058–7063, 2020.

[23] Indrajeet Yadav, Chetan D. Pahlajani, Herbert G. Tanner, and Ioannis Poulakakis. Information-sharing and decision-making in networks of radiation detectors. *Autonomous Robots*, 42(8):1715–1730, Dec 2018.