A Proposal for a Parameterized Circulating Vector Field Guidance for Fixed Wing Unmanned Aerial Vehicles

A thesis presented to

the faculty of

the Russ College of Engineering and Technology of Ohio University

In partial fulfillment

of the requirements for the degree

Master of Science

Garrett S. Clem

May 2018

This thesis titled

A Proposal for a Parameterized Circulating Vector Field Guidance for Fixed Wing Unmanned Aerial Vehicles

by

GARRETT S. CLEM

has been approved for

the Department of Mechanical Engineering

and the Russ College of Engineering and Technology by

Jay P. Wilhelm

Assistant Professor of Mechanical Engineering

Dennis Irwin

Dean, Russ College of Engineering and Technology

Abstract

CLEM, GARRETT S., M.S., May 2018, Mechanical Engineering

A Proposal for a Parameterized Circulating Vector Field Guidance for Fixed Wing Unmanned Aerial Vehicles (86 pp.)

Director of Thesis: Jay P. Wilhelm

Unmanned Aerial Vehicles (UAVs) are guided to fly along straight line obstacle free paths that connect pre-planned waypoints. Initially undiscovered obstacles encountered during flight may require waypoints to be re-planned. Obstacles could be avoided without the need to re-plan mission waypoints by implementing vector field path following in conjunction with repulsive obstacle vector fields. Repulsive vector fields that combine weighted repulsive and attractive components to provide an optimal obstacle avoidance guidance will be investigated to avoid singularities and improve path tracking performance compared to waypoint guidance.

Table of Contents

List of Figures

Figure | Page

List of Symbols

$\vec{v}$	Vector field
$\vec{v}_{conv}$	Convergence component
$\vec{v}_{circ}$	Circulation component
$\vec{v}_{tv}$	Time-varying component
G	Convergence weight
H	Circulation weight
L	Time-varying weight
q	Spatial dimension set
$\alpha_i(x_1, x_2, ...x_n, t)$	Implicit surface function
n	Number of spatial dimensions
t	Time
i	index
∇_q	Gradient with respect to spatial dimensions q
M	Gradient matrix
a	Velocity column vector
$\vec{V}$	Total field
d	Range
P	Decay weight
R	Radius
$\vec{v}_{repulsive}$	Repulsive vector field
$\vec{v}_{attractive}$	Attractive vector field
u	Speed

LIST OF ACRONYMS

UAV	Unmanned Aerial Vehicles
VF	Vector Field
UAS	Unmanned Aerial System
VFF	Virtual Force Field
TPLVF	Tangent Plus Lyapunov Vector Field
RRT*	Optimal Rapid Radom Trees
DT	Delauny Triangulation
GVF	Gradient Vector Field

1 INTRODUCTION

1.1 Motivation and Problem Statement

Unmanned aerial vehicles (UAV)s are pilotless aircraft used by military, police, and civilian communities for tasks such as reconnaissance, damage assessment, surveying, and target tracking [1,2]. Many of these tasks depend on the UAVs ability to autonomously follow a path while potentially avoiding obstacles and no-fly zones. UAV paths are typically followed by implementing heading guidance systems such as waypoint, carrot chasing, Proportional-Integral-Derivative (PID), non-linear guidance laws, or Linear Quadratic Regulator (LQR). Conventional path following guidance systems are typically not capable of avoiding obstacles without partially or completely re-planning the path. Vehicle paths are typically generated on a remote ground station and relayed to the UAV's autopilot which may be impossible under certain conditions, such as flying beyond line-of-sight. Heading guidance that accomplishes mission objectives while avoiding obstacles without the need for re-planning waypoints may be beneficial. Avoiding obstacles without path re-planning has been achieved by vector field guidance [3–7] guidance.

Vector Field (VF) guidance is a method that is mainly used for path following and can be useful for obstacle avoidance [8][WWC]. VFs can produce continuous heading vectors that can be used to guide a UAV to coverage and follow a path. Vectors are calculated by summing together convergence and circulation terms that are weighted by static scalars. Obstacles can be represented as repulsive VFs that direct the UAV away from the no-fly zone. Strictly repulsive VFs do not always route the UAV around an obstacle and can cause singularities, small regions where path following and obstacle vectors cancel. Modifying repulsive VFs to include circulation and an appropriate decay radius may be used to produce an optimal guidance. **A method for optimizing obstacle field decay radius and circulation with singularity detection is the contribution of this thesis.**

1.2 Methods Overview

The proposed research was conducted in three phases consisting of numerical simulations and indoor flight experiments. Phase I describes the construction of path following and repulsive gradient vector fields (GVFs) followed by the characterization of regions of null guidance in summed fields called singularities. Phase II investigates a numerical method for optimizing obstacle field decay radius and circulation weight which minimizes a path deviation cost function. Lastly, the optimized GVF will be implemented on an indoor flying quadcopter with fixed wing Dubin's constraints. Flight tests are then compared against simulations.

1.3 Phase I

Characterize and present a method for locating singularities in a summed GVF. Equations for constructing GVFs for path following and obstacles are presented. Target path following and obstacle fields are summed together and the presents of singularities is discussed. A method for locating singularities numerically in a summed field is presented.

1.4 Phase II

Determine a combination of circulation and decay radius for a circular obstacle GVF that produces an optimized obstacle avoidance. A UAV modeled as a Dubin's vehicle is used to demonstrate GVF guidance for converging and following a straight path. A worst case collision scenario is presented along with several GVF guidance solutions. The path deviation of each GVF guidance is

A worst case collision scenario where an obstacle lies centered on the path is presented and guidance for various GVF weights

Obstacle field decay radius and circulation weight is determined by searching a large parameter space as well as numerically. The modified GVF will be compared against a static and strictly repulsive GVF. Deviation from the target path while avoiding the obstacle is used to compare the optimized GVF with unmodified GVF.

1.5 Phase III

Demonstrate optimized GVF guidance on multirotor UAV flying with fixed wing turn-rate constraints. The modified GVF developed in Phase II will be implemented on a crazyflie 2.0 quadcopter flying in an indoor environment. Dubin's turn rate constraints will be applied in order to emulate the behavior of a fixed wing UAV. Deviation from a planned path while avoiding a circular obstacle will be compared to simulation results.

1.6 Summary of Objectives

Each phase consists of an **objective** that was accomplished by executing several *tasks*. Completion of all objectives and phases will result in the final <u>deliverable</u>.

Phase I Objective: Demonstrate and locate singularities in a summed gradient vector field
Tasks:

1. *Construct GVF equations from literature*

2. *Evaluate scenarios where singularities are expected*

3. *Characterize location of singularities*

Phase II Objective: Determine combination of repulsive gradient vector field circulation and decay radius that minimizes path deviation
Tasks:

1. *Define obstacles in terms of UAV turning radius*

2. *Determine combination of GVF decay radius and circulation weight that produces minimal path deviation*

Phase III Objective: Validate modified gradient vector field guidance with indoor quadrotor experiments

Tasks:

1. *Build quadcopter and assemble testing hardware*
2. *Program guidance and control systems into Python ground station*
3. *Repeat simulations performed in Phase II on quadcopter and compare results*

Deliverable: An optimized GVF for avoiding circular obstacles that lie along a target path

2 Literature Review

2.1 Literature Review Introduction

fill out

2.2 Unmanned Aerial Vehicles

Unmanned aerial vehicles (UAV)s are pilotless aircraft used by military, police, and civilian communities for tasks that may put pilots of conventional manned aircraft in harms way [9]. Tasks can be carried out by a single UAV or in cooperation with another air [10–12], ground [13], or marine vehicles. Using UAVs has several advantages over manned aircraft consisting of low operating cost, reduced risk to human operators, and the ability to perform mundane and repetitive tasks autonomously without heavy human interaction. In general, UAVs are categorized into fixed wing and rotor craft varieties [14] that range in size, payload, and flight time capabilities. Fixed wing UAVs (Figure 2.1a) are typically used for tasks that require longer flight times and larger payloads, such as cameras and cargo. Multirotor UAVs (Figure 2.1b), in general, have lower payload capabilities compared to fixed wing UAVs, however have the ability to hover and have a small turning radius making them highly maneuverable.

(a) (b)

Figure 2.1: Fixed wing (a) and multirotor (b) UAVs

Weather it is a fixed wing or rotorcraft, UAVs typically accomplish tasks autonomously by following a pre-planned path. Flight paths are typically generated off-line at dedicated ground station and relayed to the UAV over radio.

2.2.1 Ground Stations

Ground stations are the hardware and software framework used to configure UAV settings, plan missions, and collect mission data. Commercial open source ground stations such as qground control depend on a human operator's knowledge of the environment to plan an obstacle free path. Takeoff, landing, and emergency return-to-home locations are designated in safe clearings capable of accommodating the vehicle. Other tasks such as area surveying and loitering can be assigned at certain points along the mission path. An example of a mission consisting of taking off, surveying, and landing is shown in Figure 2.2

Figure 2.2: Ground station software planning a waypoint based mission

These paths are commonly represented as a series of finite waypoints in commercial UAV autopilots such as the Piccolo [15], Kestral [16], and Pixhawk [17].

2.2.2 Autopilot

The UAV autopilot is responsible for controlling a pre-planned path and maintaining vehicle stability while under the influence of external wind disturbances. Stable flight while path following is accomplished by implementing feed-back control, navigation, and guidance systems. Feed-back refers to the closure of an open-loop control system which allows a reference error to be calculated between the desired state of the UAV, the reference, and the current state of the UAV. Reference error is used to calculate the necessary actuator output required to modify the vehicles attitude and position while preventing unbounded oscillation. Attitude and position feed-back is provided by the navigation system by sampling on-board sensors such as global position system (GPS) and inertial measurement units (IMUs). Filtering and fusing noisy data from multiple sources is often accomplished through estimation techniques such as the Kalman filter. The guidance system directs the UAV with a commanded heading towards a desired goal, such as a waypoint or path. A high level overview of the autopilots systems can be seen in Figure 2.3.

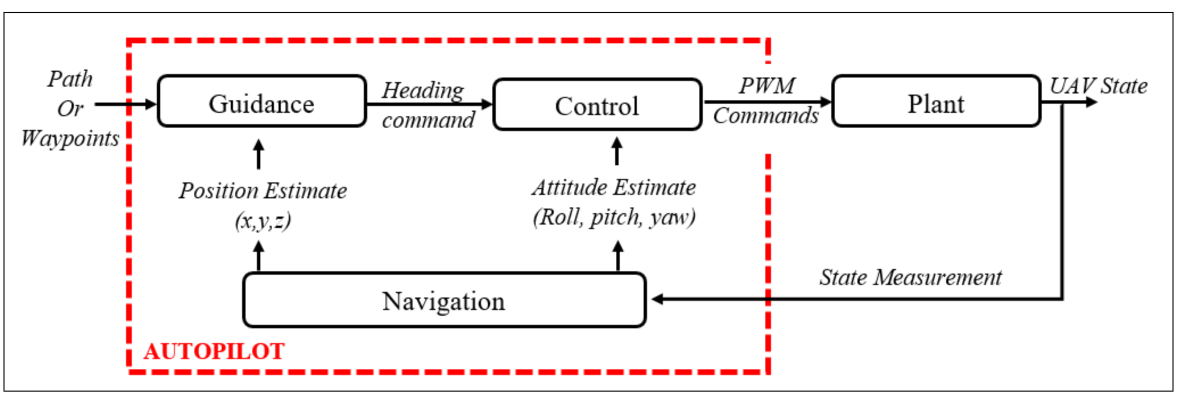


Figure 2.3: Autopilot's Navigation, Guidance, and Control Architecture

2.3 Dubins Vehicle Model

The dynamics of UAVs are often simplified when simulating guidance systems by modeling the UAV as a Dubin's vehicle [3, 4, 18–20]. It is assumed that the autopilot's control system is capable of maintaining stability, speed u, and can turn the vehicle at a fixed turn rate $\dot{\theta}$. The position of the UAV $\overrightarrow{X}$ at time t is calculated from the integral of the velocity vector $\overrightarrow{U}$, Equation 2.2. Heading θ is an input from a guidance system. Here the turnrate of the UAV is restricted to 20 degrees per second.

$$\overrightarrow{U}(t) = u \begin{bmatrix} cos(\theta(t)) \\ sin(\theta(t)) \end{bmatrix} \tag{2.1}$$

$$\overrightarrow{X}(t) = \overrightarrow{U}dt + \overrightarrow{X}(t-1) \tag{2.2}$$

$$\dot{\theta} \leq 20deg/s \tag{2.3}$$

2.4 UAV Guidance

On-board guidance systems attempt to minimize the lateral error to the path by commanding a heading pointing to the path. Guidance methods for following a pre-planned path include geometric methods such as carrot chasing [21] and control techniques such as proportional-integral-derivative (PID), non-linear guidance laws, and linear quadratic regulator (LQR) [22]. Due to traditional guidance method's dependence on a path planner to construct an obstacle free and flyable path, these methods often lack a mechanism to avoid new obstacles. Re-planning and relaying a new obstacle free path may be impossible under certain conditions, such as flying beyond line-of-sight. It would be beneficial to include obstacle avoidance into a UAVs guidance system to remove the need to communicate with the ground station or use an on-board path planner which may be accomplished with potential field or vector field.

2.4.1 Potential Field

Potential field is based on the principle of driving a point masses state from a high potential to a globally low potential using artificial attractive and repulsive forces [23]. Attractive forces represent a goal, such as a waypoint, that directs the point mass in the goal's direction. Obstacles can be represented as repulsive forces that act locally to push the point mass away. An example of a point mass directed by potential field from a high potential to a globally minimum potential while avoiding an obstacle is shown in Figure 2.4. Potential field is also capable of acting as a path and trajectory planning algorithm [24], possibly eliminating the off-board path planner.

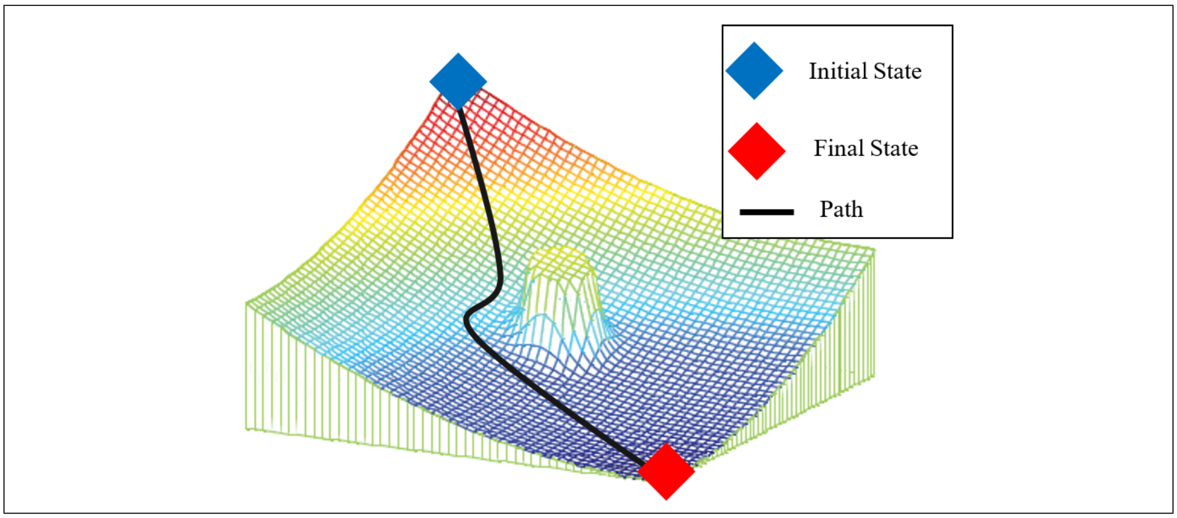


Figure 2.4: Single Obstacle Potential Field Gradient [25]

An implementation of potential field on a mobile ground robot equipped with ultrasonic sensors for real-time obstacle detection can be found in [26–28]. The differential drive robot was guided to a goal with the guidance $\overrightarrow{R}$ while avoiding obstacles. The robot was attracted towards a goal with constant magnitude force $\overrightarrow{F_t}$ located at (x_t, y_t) and a distance d_t from the robot. In the immediate area of the robot, an active window exists which records integer certainty values inside discrete cells. Cells containing an obstacle provide a repulsive force $\overrightarrow{F_{i,j}}$ opposite in direction to the line-of-sight from vehicle to cell location (x_i, y_j), where (i, j) represents the cell index, F_{cr} is a constant repulsive force, W the vehicle's width, $C_{i,j}$ a cell's certainty, and $d_{i,j}$ the distance to the center of the cell with respect to robots center.

$$\overrightarrow{R} = \overrightarrow{F_r} + \overrightarrow{F_t} \tag{2.4}$$

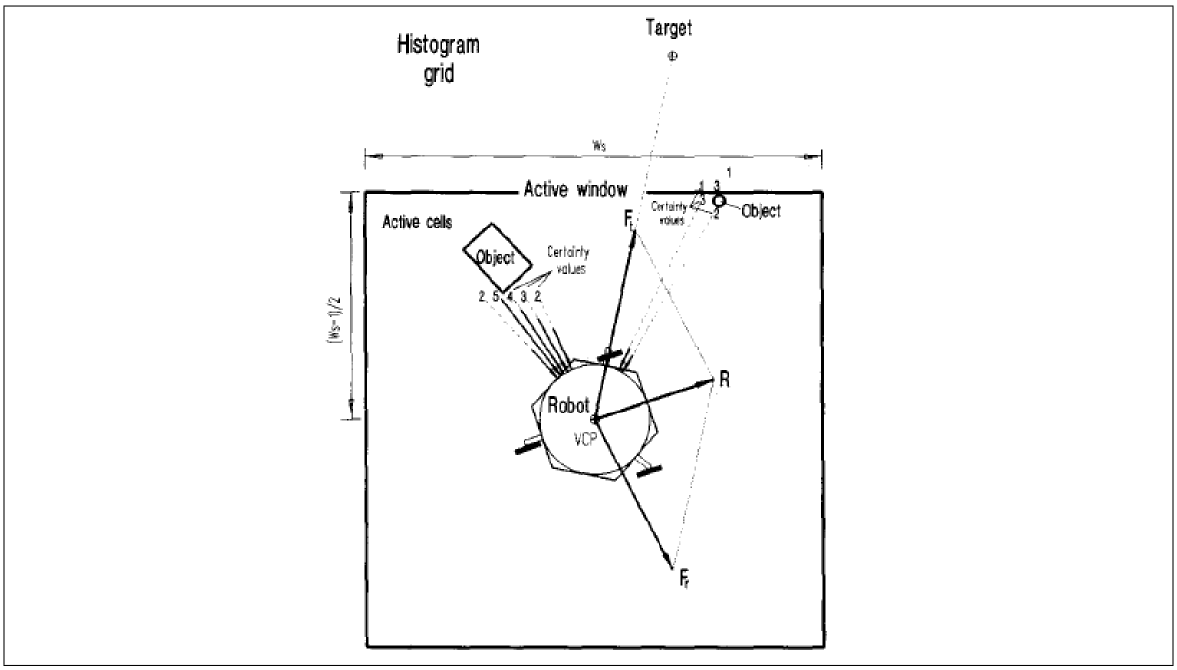


Figure 2.5: Virtual force field histogram acting on a mobile robot [27]

$$\overrightarrow{F_{i,j}} = \frac{F_{cr} W^n C_{i,j}}{d_{i,j}^n} \left(\frac{x_i - x_0}{d_{i,j}} \hat{x} + \frac{y_i - y_0}{d_{i,j}} \hat{y} \right) \tag{2.5}$$

The total repulsive force exerted on the robot is determined by summing the active cells, shown in Equation 2.6

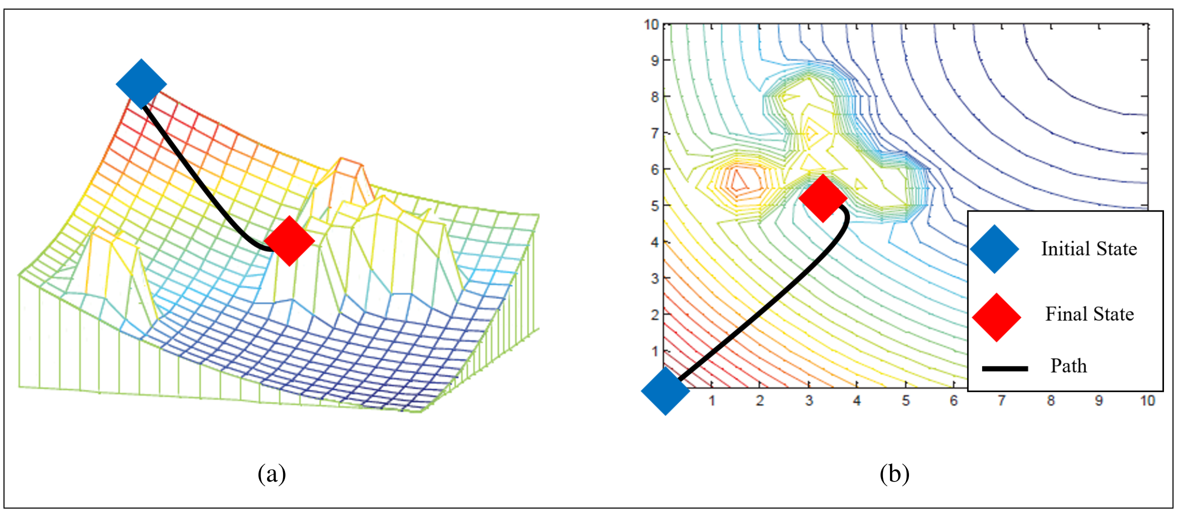
$$\overrightarrow{F_r} = \sum_{i,j} \overrightarrow{F_{i,j}} \tag{2.6}$$

$$\overrightarrow{F_t} = F_{ct}\left(\frac{x_t - x_0}{d_t}\hat{x} + \frac{y_t - y_0}{d_t}\hat{y}\right) \tag{2.7}$$

Summing together attractive and repulsive forces produce a vector $\overrightarrow{R}$ that can be used for heading guidance, shown in Equation 2.8.

$$\overrightarrow{R} = \overrightarrow{F_r} + \overrightarrow{F_t} \tag{2.8}$$

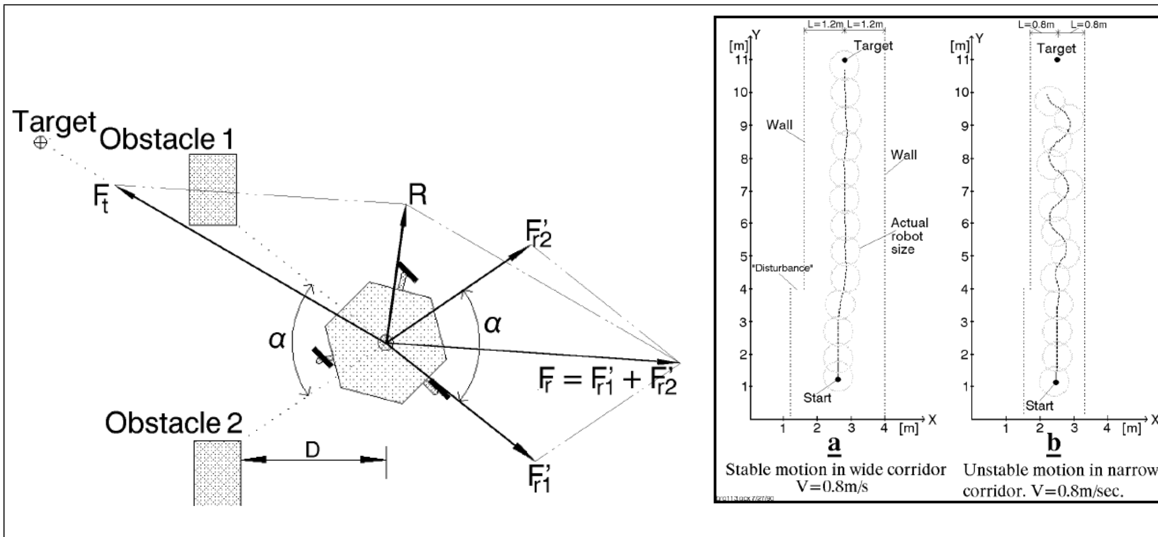
Major drawbacks to potential field were identified in [28] consisting of local minimum and oscillations in corridors. Local minimum are solutions in the potential field that are caused by the close spacing of obstacles which may result in a well that can trap the system into a local stable point prior to reaching the global minimum. A scenario similar to that shown in Figure 2.4 with more obstacles results in a trap situation where the state settles into a well on the gradient, shown in Figure 2.6. Additionally, closely spaced obstacles may also be difficult to pass between, shown in Figure 2.7a. Oscillations can also be experienced near obstacles or in narrow passages at high speeds, shown in Figure 2.7b.



(a) (b)

Figure 2.6: Potential Field Local Minimum [25]

Proposed solutions to local minimum include object clustering and virtual waypoint method [25], virtual escaping route [29], and use of navigation functions [30]. Oscillations in potential field were addressed in [31] and [32].



(a) (b)

Figure 2.7: Potential Field Local Minimum [25]

Navigation functions [30] and obstacle clustering [25] have been used to prevent local minimums in potential field. Navigation functions relate kinematic constraints to the gradient potential to produce a bounded and local minimum free solution [24]. Clustering closely spaced obstacles into a single and equally repulsive obstacle prevents local minimum from forming, shown in Figure 2.8.

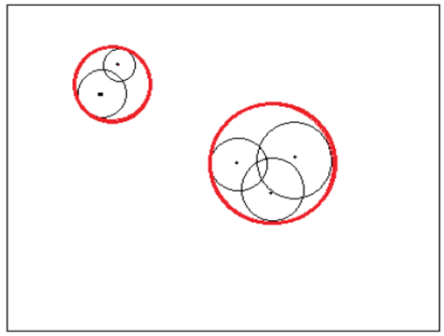


Figure 2.8: Obstacle Clustering [25]

Potential Field's ability to avoid obstacles and combine path planning, trajectory planning, and control into a single computationally inexpensive system makes it an attractive motion control system for robots seeking a singular point, even with the limitations discussed in [28]. Unlike the mobile ground robots in [26], fixed wing UAVs must maintain a minimum forward velocity, have limited turning radius, and cannot converge to a single point. Vehicles with velocity and turn rate constraints may not return to a pre-planned path once the obstacle has been avoided, shown in Figure 2.9. Methods that direct a UAV to a path instead of a discrete point has been achieved with Lyapunov and gradient vector fields.

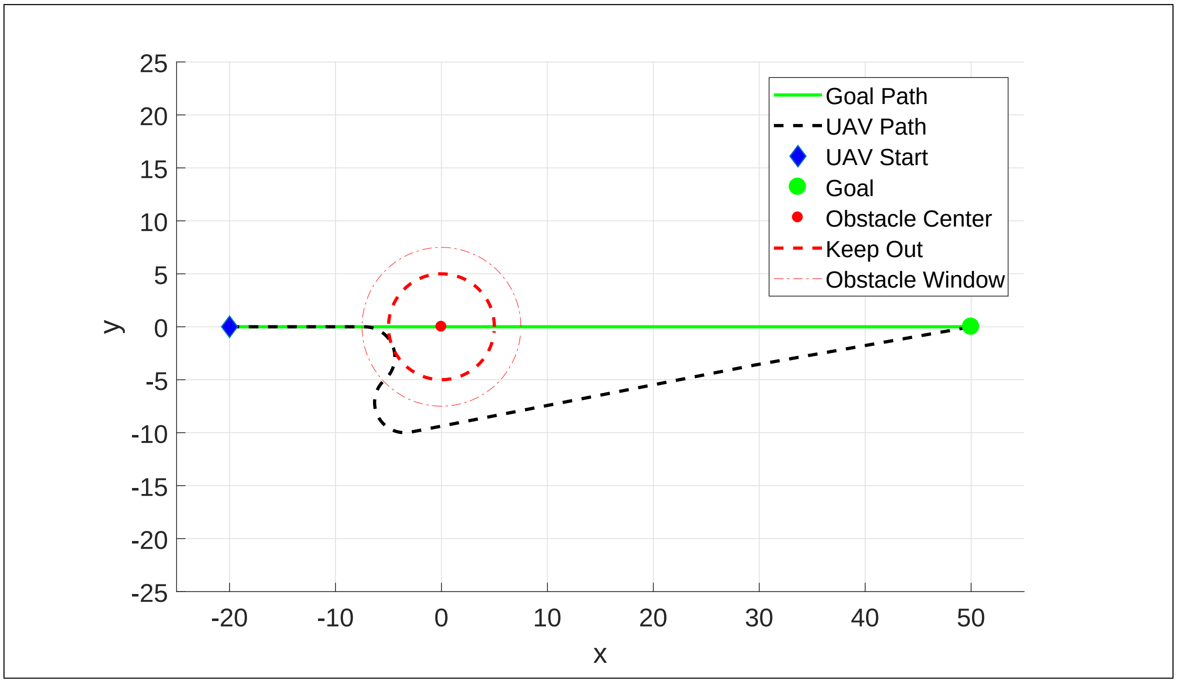


Figure 2.9: UAV avoiding obstacle with VFF Guidance

2.4.2 Lyapunov Vector Fields

Lyapunov vector fields for converging and following straight and circular paths were described in [18]. For converging and following a straight path, a guidance vector χ^d is determined in Equation 2.9, where χ^∞ is the course approach angle, y is the lateral distance to the path, and k is a positive constant that determines the rate of transition between convergence and following. An example of a Lyapunov vector field converging and following a straight line is shown in Figure 2.10a.

$$\chi^d(y) = -\chi^\infty \frac{2}{\pi} \tan^{-1}(ky) \tag{2.9}$$

For converging and following a circular path, a guidance vector χ^d is determined in Equation 2.10, where γ is the UAVs angular position with respect to the circle, r is the paths radius, d is the distance from the circles center, and k is a positive constant that determines the transition behavior. An example of a Lyapunov vector field for converging and following a circular path is shown in Figure 2.10b.

$$\chi^d(d) = \gamma - \frac{\pi}{2} - \tan^{-1}\left(k\frac{d-r}{r}\right) \tag{2.10}$$

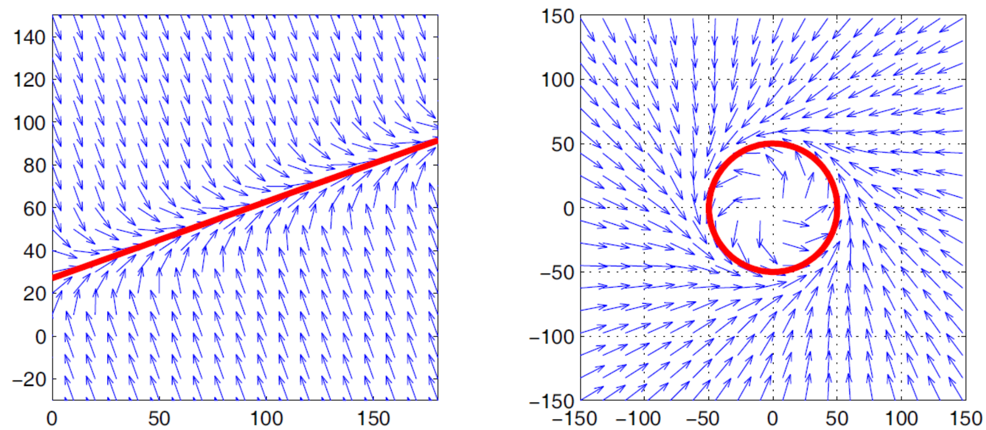


Figure 2.10: Lyapunov vector field for straight line and circular primitives [18]

Straight and circular path vector fields can be selectively activated throughout flight to form more complex paths, shown in [18–20, 33] and Figure 2.11. Each path primitive has a vector field associated with it and determining which field to use can be approached in two different ways. Fields from all of the primitives can be summed together similar to the attractive and repulsive forces in potential field. Second, fields can be selectively activated and deactivated based on the position of the UAV. Summing together vector fields, as pointed out in [18], can result in several problems including dead zones, sinks, and singularities. Selectively activating each vector field as a UAV nears waypoints was used in [18–20, 33].

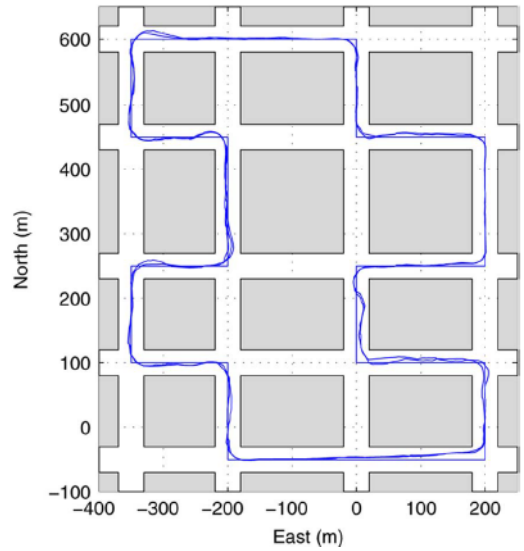


Figure 2.11: Straight path following in urban environment [18] using Lyapunov Vector Field

Lyapunov Vector field construction for curved paths was presented in [4] and is shown in Figure 2.12. Constructing a Vector Field for an arbitrary curve may allow for more complex paths and could eliminate the need for switching between primitives.

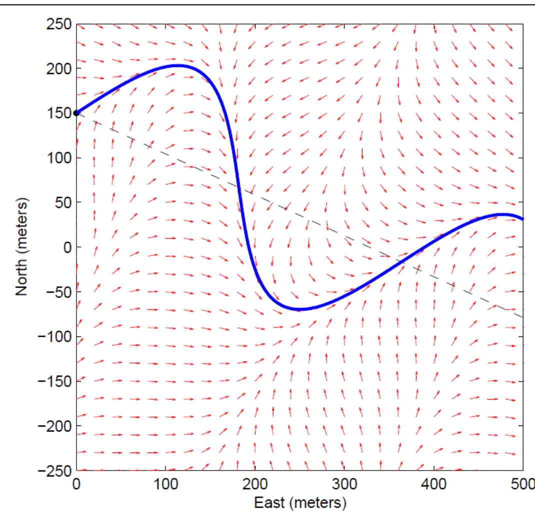
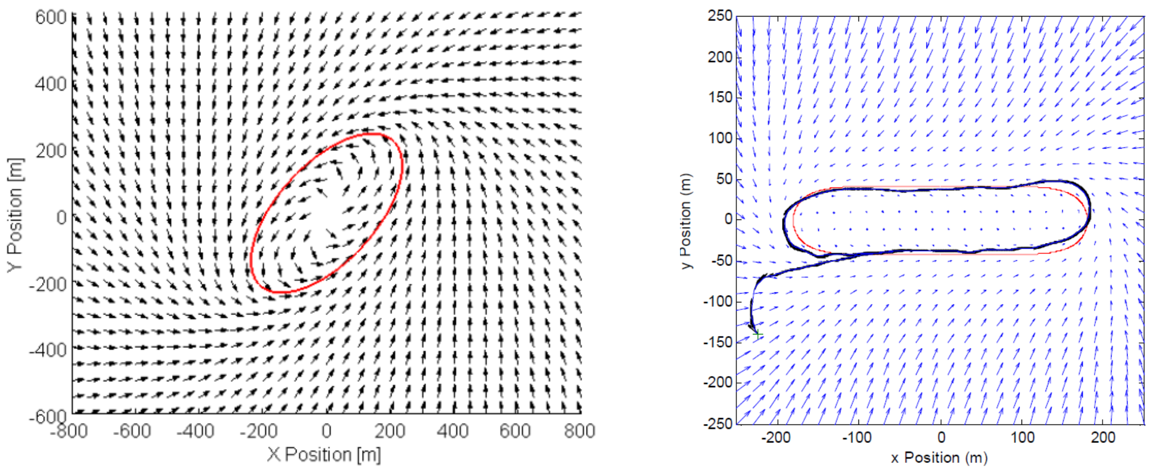


Figure 2.12: Lyapunov vector field approach curved path asymptotically [4]

Primitive circular vector fields were modified in [3, 34] via non-linear coordinate transformations to produce elliptical 2.13a, or racetrack 2.13b, fields. Transforming the circular field as a function of a Kalman filter's covariance matrix when sensing an uncertain target was investigated in [3].



(a) (b)

Figure 2.13: Elliptical VF produced by non-linear coordinate transformations a) [3] and b) [34]

Target tracking Tangent Plus Lyapunov Vector Field (TPLVF) was introduced in [35] that produced shorter paths compared to Lyapunov alone. Outside of the standoff circle, tangent vectors provided the shortest distance to a standoff circle. Inside the standoff circle, no tangent lines exist and Lyapunov was used in its place. Figure 2.14 shows the difference in paths taken for Lyapunov and tangent vector fields outside the standoff circle. The TPLVF was later used for path planning to avoid obstacles in [36] while [37] constructed a tangent vector field for curved paths.

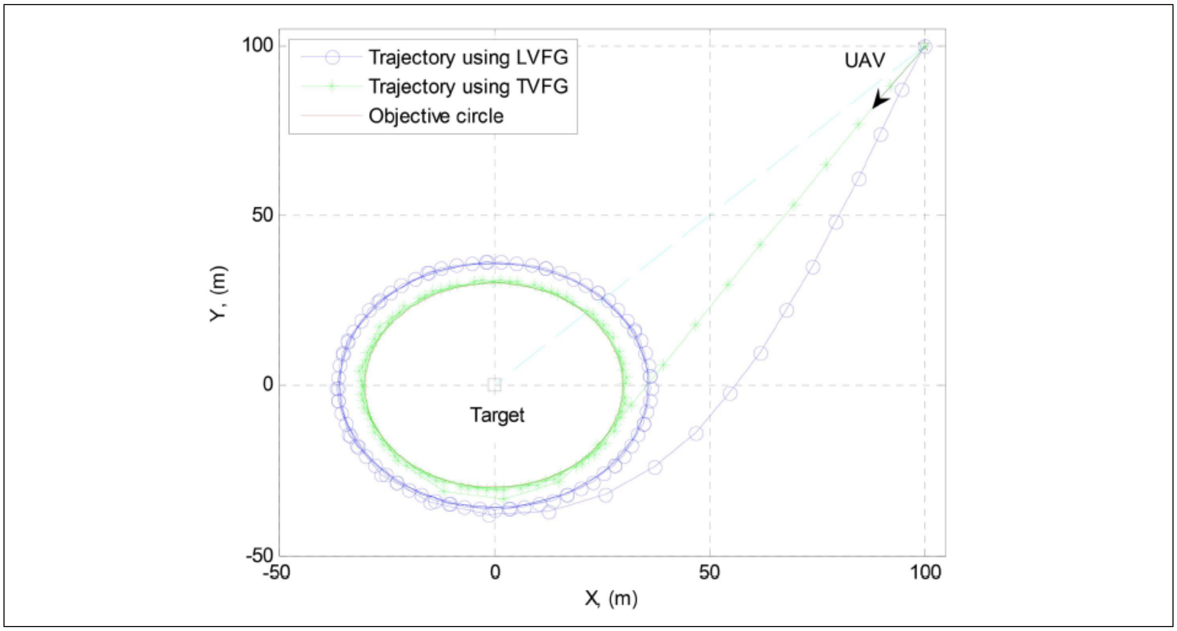


Figure 2.14: Tangent plus lyapunov vector fields for shortest path target tracking [36]

2.4.3 Path Planning Vector Fields

Another use of vector fields is a high level specification for heuristic path planning algorithms [38]. An optimal Rapid Random Trees (RRT*) algorithm used a vector field as a guide to explore the configuration space of the UAV for an obstacle free path. Branches extend from the root, or initial location of the UAV, randomly throughout the map with a finite deviation from the initial vector field. When a branch encounters an obstacle it is trimmed and no longer explored. The path of minimum cost, or least distance, is selected for the UAV to use as a reference path. An example of the algorithm is shown in Figure 2.15.

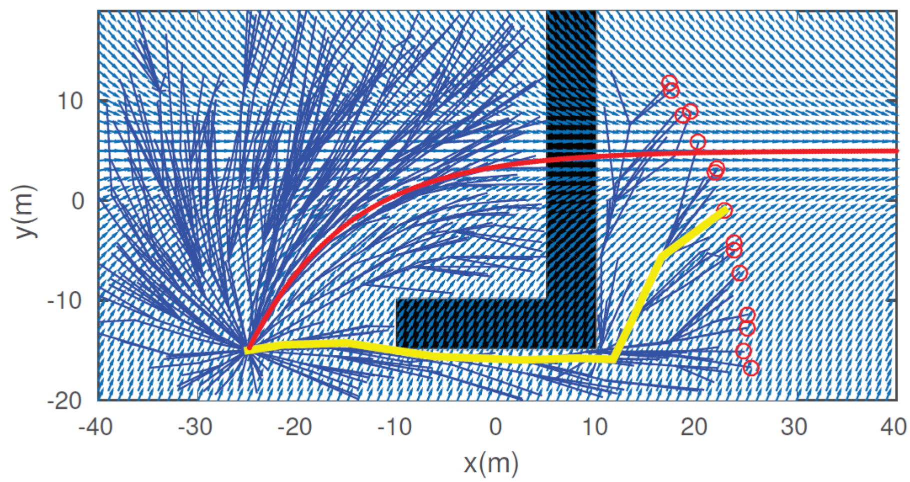


Figure 2.15: RRT* path planner with a VF used as a task specification [38]

A VF was constructed inside a configuration space with edges defined by Delauny triangulation (DT) in [39]. A simulation of a robot traversing a vector field inside a set of DTs can be seen in Figure 2.16. Vector fields designed to stay inside a region of DTs may be used with optimal path planning algorithms for navigating urban environments [40].

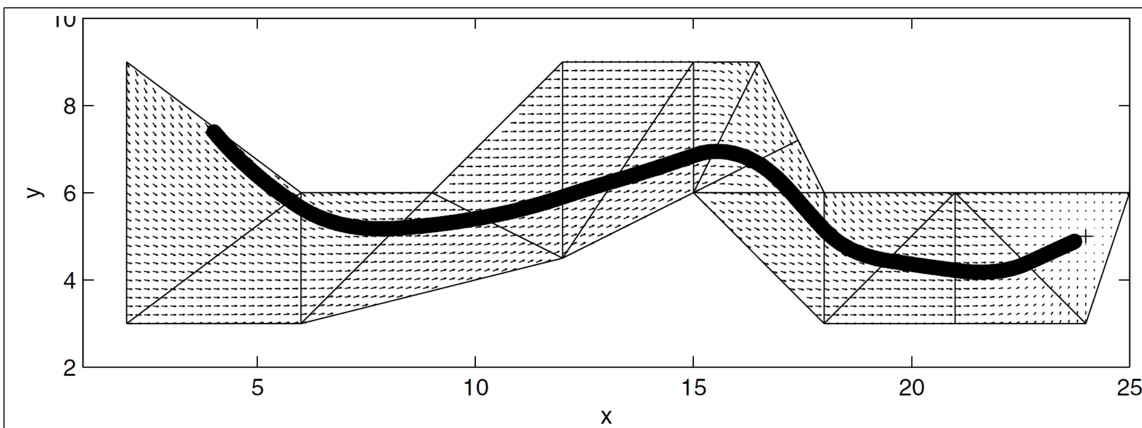


Figure 2.16: Vector field within a set of delaunay triangles [39]

So far all of the vector field methods discussed have avoided obstacles by planning paths around them. Paths are typically calculated at the ground station and if communication is lost a new path may not be relayed to a UAV encountering a new obstacle.

A possible solution is using vector fields to provide a repulsive force such as that seen in [8, 41] [wwc].

2.4.4 Gradient Vector Field

The Gradient Vector Field (GVF) method produces a similar field, however has several advantages over LVFs. GVF produces an n-dimensional vector field that converges and circulates to both static and time varying paths [5]. Additionally, convergence, circulation, and time-varying terms that make up the GVF are decoupled from each other allowing for easy weighting of the total field [6]. GVFs converge and circulate at the intersection, or level set, of $n-1$ dimensional implicit surfaces ($\alpha_i : \mathbb{R}^n \to \mathbb{R} | i = 1, ..., n-1$). The integral lines of the field are guaranteed to converge and circulate the level set when two conditions are met: 1) the implicit surface functions are positive definite and 2) have bounded derivatives. Consider the space with dimensions in set **q**:

$$\mathbf{q} = \left[x_1, x_2, ..., x_n \right] \tag{2.11}$$

The total vector field $\overrightarrow{V}$ is calculated by:

$$\overrightarrow{V} = G\nabla V + H \wedge_{i=1}^{n-1} \nabla_q \alpha_i - LM(\alpha)^{-1} a(\alpha) \tag{2.12}$$

or in component form:

$$\overrightarrow{V} = \overrightarrow{V}_{conv} + \overrightarrow{V}_{circ} + \overrightarrow{V}_{tv} \tag{2.13}$$

where $\overrightarrow{V}_{conv}$ produces vectors that converge to the path, $\overrightarrow{V}_{circ}$ produces vectors that circulate the path, and $\overrightarrow{V}_{tv}$ is a feed-forward term that produces vectors accounting for a time varying path. The scalars G,H, and L weight convergence, circulation, and time varying components respectively.

Convergence is calculated by:

$$\overrightarrow{V}_{conv} = G\nabla V \tag{2.14}$$

where scalar G is multiplied by the gradient of the definite potential function V:

$$V = -\sqrt{{\alpha_1}^2 + {\alpha_2}^2} \tag{2.15}$$

Circulation is calculated by taking the wedge product of the gradient:

$$\overrightarrow{V}_{circ} = \wedge_{i=1}^{n-1} \nabla_q \alpha_i \tag{2.16}$$

In the case of $(n = 3)$ the wedge product simplifies as the cross product:

$$\overrightarrow{V}_{circ} = \nabla_q \alpha_1 \times \nabla_q \alpha_2 \tag{2.17}$$

The feed-forward time-varying component is calculated by:

$$\overrightarrow{V}_{tv} = M^{-1} a \tag{2.18}$$

where,

$$M = \begin{bmatrix} \nabla\alpha_1^T \\ \nabla\alpha_2^T \\ (\nabla\alpha_1 \times \nabla\alpha_2)^T \end{bmatrix} \tag{2.19}$$

$$a = \begin{bmatrix} \frac{\partial\alpha_1}{\partial t} & \frac{\partial\alpha_2}{\partial t} & 0 \end{bmatrix}^T \tag{2.20}$$

In [5–7] GVFs were constructed to control the velocity $\dot{q}$ of a holonomic robot by $\dot{q} = h$. Constant speed u was controlled by calculating the weighting scalars G,H, and L that maintained the condition $||\dot{q}|| = u$. Other studies normalized the vector $\overrightarrow{V}$ and used it as a heading guidance while assuming velocity is held constant by the autopilot [42][wwc].

Assuming velocity is controlled by a separate system frees up the vector field scalars to modify field behavior for other applications, such as obstacle avoidance.

The standoff tracking and avoidance scenario presented in [wwc] used GVF as a heading guidance and static GVF weights to specify high level guidance behavior. A fixed UAV was tasked with loitering around a slow moving ground target while avoiding obstacles. A circular attractive time-varying vector field $\overrightarrow{V}_p$ was attached to a moving ground target and summed with repulsive obstacle vector fields $\overrightarrow{V}_o$ centered at the obstacles to provide the guidance $\overrightarrow{V}$ in Equation 2.21.

$$\overrightarrow{V} = \overrightarrow{V}_p + P\overrightarrow{V}_o \tag{2.21}$$

The strength of repulsive obstacle fields were weighted by the hyperbolic tangent decay function $P(d)$ in Equation 2.22, where d is the range to the obstacle and R is the radius of the decay, shown in Figure 2.17

$$P = -R\frac{tanh(2\pi d - \pi) - 2}{2} \tag{2.22}$$

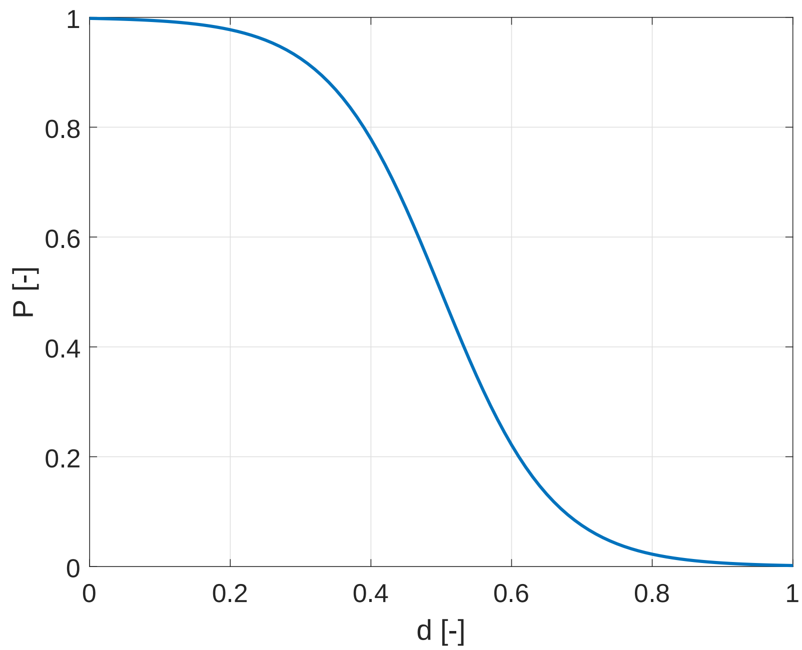


Figure 2.17:

The performance of LVF [21] and GVF [5–7] were compared for their cross track error with respect to the loiter circle in [wwc]. GVF was shown to have less cross track error compared to LVF due to the compensation for a time-varying vector field attached to the moving target. The path of the fixed wing UAV tracking a slow moving ground target while avoiding static obstacles is shown in Figure 2.18

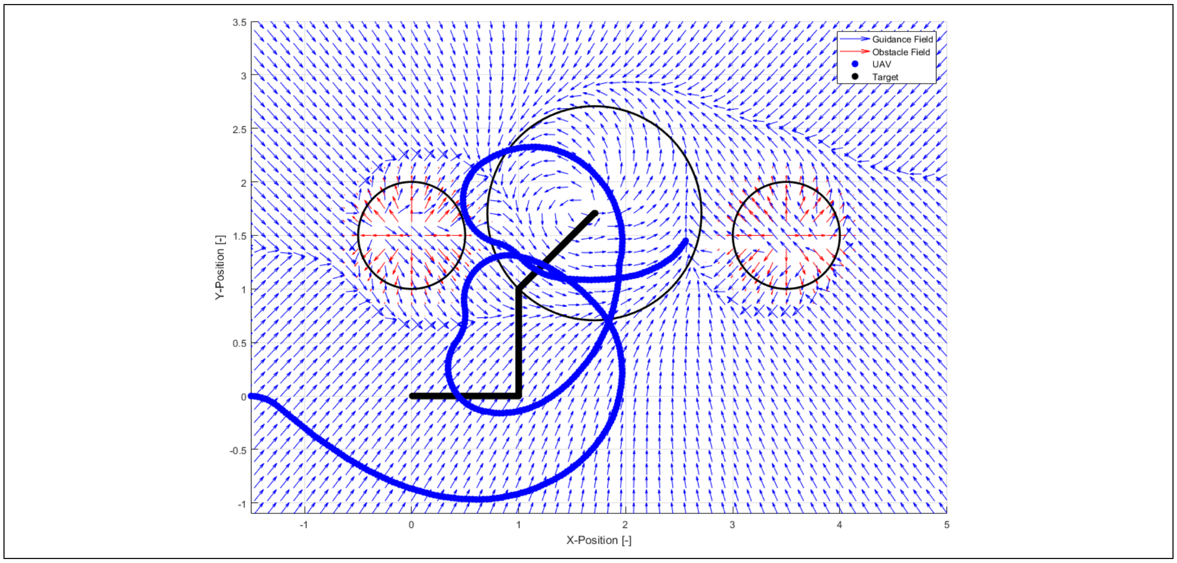


Figure 2.18: Place holder image of UAV following ground target [**?**]

Summing attractive and repulsive vector fields may result in null guidance where the fields cancel, providing no guidance. The presence of singularities were mentioned briefly in [18] and observed in [8]. For fixed wing UAVs the lack of guidance may prevent the UAV from avoiding an obstacle, while multi-rotor UAVs may end up in a trap situation. Singularities may be present at any location where a goal field and obstacle field are of equal strength. Detecting singularities and modifying the GVF for an improved obstacle avoidance is the contribution of this research.

2.5 Literature Review Summary

UAVs are versatile tools that can be used for remote data collection without putting a pilot in harms way. Tasks are typically accomplished by following a series of waypoints that lie along an obstacle free and flyable pre-planned path. When unplanned or previously unknown obstacles are discovered along the UAVs planned path, a new obstacle free and flyable path may have to be generated which could be impossible if the ground station is not reachable. Methods such as potential field and vector field have been used to avoid obstacles without re-planning mission waypoints. Potential field is not ideal for fixed wing UAVs since it converges to a singular point whereas Lyapunov and gradient vector fields converge and follow paths. Summing paths and negating field weights can be used for obstacle avoidance, however have so far been used only as high level specification of guidance behavior. Gradient vector field allows for easy weighting of individual convergence and circulation terms therefore is the best candidate for providing an optimized avoidance field.Determining repulsive field weigths and decay radius has not been optimized for obstacle avoidance. Additionally, characterization of vector field singularities in a summed field has yet to be addressed in literature due to most methods avoiding summation entirely. The contribution of this research is to characterize GVF singularities and provide an optimized decay radius and circulation weight for repulsive fields that minimize path deviation.

3 Methodology

3.1 Introduction to Methodology

A real-time circular obstacle avoidance guidance will be achieved by summing together a path following and obstacle avoidance vector field with optimized decay and circulation weights. Singularities in summed vector fields are defined and a method for numerically locating their position is presented in Phase I and it is shown how adding circulation to repulsive GVFs may remove singularities from the UAVs path. Phase II investigates a method for selecting the combination of repulsive GVF decay radius and circulation weights that minimizes a path deviation cost function. The optimized GVF guidance is implemented on an indoor multirotor UAV operating under fixed wing constraints in Phase III.

3.2 Phase I: Gradient vector field singularity detection

The objective of Phase I is to characterize and present a method for locating singularities in a summed gradient vector field. Phase I consists of calculating guidance for converging and following a straight path using the GVF method in literature. A repulsive GVF is constructed for avoiding circular obstacles along the path by modifying the GVF's decay radius, convergence, and circulation weights. Summing the attractive and repulsive GVFs results in guidance that directs the UAV along the planned path while pushing away from the obstacle. Regions in the summed guidance where the path following and obstacle guidance directly oppose each other can result in vectors of zero length,called singularities. A method for identifying the location of singularities is presented along with a method for mitigating them.

3.2.1 Path Following Vector Field Guidance

Guidance for converging and following a time-invariant path using GVF guidance is achieved by summing together convergence and circulation terms that are multiplied by scalars G and H respectively, shown in Equation 2.13. The potential function V in Equation 2.15 decreases asymptotically to null when approaching the target path and therefore the convergence vector begins to decrease as well [5]. As the potential function decreases, the circulation term begins to dominate the guidance, promoting path following. How close to the target path the transition between convergence and circulation depends on the scalar weights G and H respectively. The total vector $\overrightarrow{V}_p$ represents the non-normalized attractive path following vector field comprised of both convergence and circulation terms. The shape of the target path that the vectors converge and follow depend on the specification of the implicit 3-dimensional surface functions α_1 and α_2. Intersecting two planes can be used to generate a GVF that converges and follows a straight path. The vertical plane, described in Equation 3.1, at angle δ with respect to the paths $y - axis$, is intersected with a horizontal plane at constant height z, Equation 3.2

$$\alpha_1 = cos(\delta)x + sin(\delta)y \tag{3.1}$$

$$\alpha_2 = z \tag{3.2}$$

The gradient of the potential function, ∇V, for calculating the path following convergence term is shown in Equation 3.3.

$$\nabla V = -\frac{1}{2(\sqrt{\cos^2(\delta)x^2 + 2\cos(\delta)\sin(\delta)xy + \sin^2(\delta)y^2})}\begin{bmatrix} 2x\cos^2(\delta) + 2\cos(\delta)\sin(\delta)y \\ 2y\sin^2(\delta) + 2\cos(\delta)\sin(\delta)x \\ 2 \end{bmatrix} \tag{3.3}$$

Circulation is calculated by the cross product of the surface function gradients, which evaluates to that shown in Equation 3.4.

$$\overrightarrow{V}_{circ} = \begin{bmatrix} sin(\delta) \\ -cos(\delta) \\ 0 \end{bmatrix} \tag{3.4}$$

The vector field $\overrightarrow{V}_p$ described will converge and circulate a straight path at the intersection of surfaces α_1 and α_2, depicted in Figure 3.1.

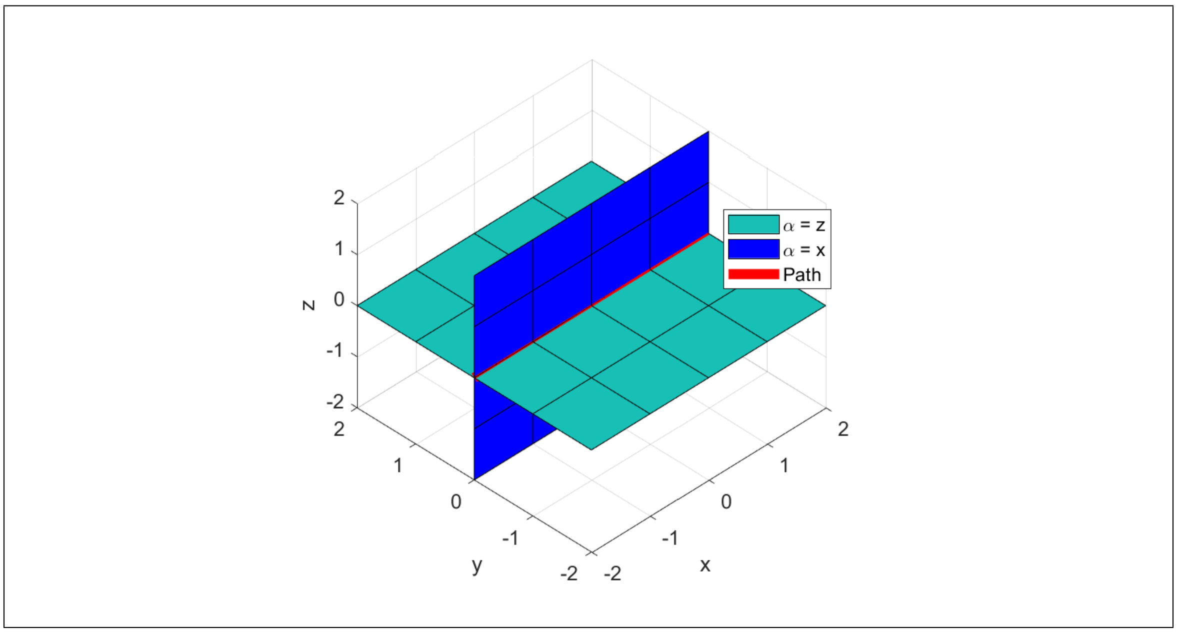


Figure 3.1: Intersection of planes defined by implicit surface functions

Prior to using the path following guidance $\overrightarrow{V}_p$, it is normalized, denoted as $\overrightarrow{V}_{||P||}$, to have a magnitude $||\overrightarrow{V}_p|| = 1$. The reason for normalizing the summed path following vector field is threefold. First, the vector is used as a heading controller only, therefore the angle of the vector is the information required by the autopilot. Second, the normalized vectors result in quiver plots with equal density arrows making the field easier to visualize. Lastly, normalizing the path following vector $\overrightarrow{V}_p$ fixes the length of the vector allowing for prediction of singularity location after summing the field, which will be discussed in the next section. Before discussing singularities, the obstacle field is introduced.

$$\overrightarrow{V}_{||P||} = \frac{\overrightarrow{V}_p}{||\overrightarrow{V}_p||} \tag{3.5}$$

To produce guidance for following a path and avoiding an obstacle, a repulsive obstacle vector field $\overrightarrow{V}_{||O||}$ needs to be constructed and summed with the normalized path following guidance $\overrightarrow{V}_{||P||}$. The repulsive vector field is multiplied by a decay function P which limits the influence of the obstacle to a finite range and will be discussed after the avoidance field equations are presented. The sum of the two guidances is represented by $\overrightarrow{V}_g$ and is shown in Equation 3.6

$$\overrightarrow{V}_g = \overrightarrow{V}_{||P||} + P\overrightarrow{V}_{||O||} \tag{3.6}$$

Constructing the obstacle avoidance vector field $\overrightarrow{V}_{||O||}$ will now be discussed.

3.2.2 Constructing an Avoidance Vector Field

A circular avoidance vector field can be constructed in a way similar to that of the path following field in the previous section. What differentiates the obstacle field from the path following field is that the individual convergence $\overrightarrow{V}_{conv}$ and circulation $\overrightarrow{V}_{circ}$ components are normalized prior to summing to result in the obstacle field $\overrightarrow{V}_o$. The benefit of normalizing each field component before multiplying by their respective scalars G and H prior to summing is to produce an obstacle field with uniform behavior as distance from the obstacle increases. In short, normalizing each component allows both convergence and circulation terms to be present in the obstacle guidance at larger distances. Additionally, negative convergence weights will be used to produce vectors that diverge away from the path. The obstacle vector field is constructed using the normalized component Equation 3.7 with obstacle convergence and circulation weights G_o and H_o respectively.

$$\vec{V}_o = G_o \frac{\vec{V}_{conv}}{\|\vec{V}_{conv}\|} + H_o \frac{\vec{V}_{circ}}{\|\vec{V}_{circ}\|} \tag{3.7}$$

A circular avoidance vector field with radius r centered at (x_c, y_c) is constructed by intersecting a cylinder, Equation 3.8, and a plane Equation 3.2.

$$\alpha_1 = (x - x_c)^2 + (y - y_c)^2 - r^2 \tag{3.8}$$

Convergence is calculated by the gradient of the potential function 3.3, which when simplified evaluates to

$$\nabla V = A\vec{B} \tag{3.9}$$

where

$$A = \frac{-1}{\sqrt{\bar{x}^4 + \bar{y}^4 + 2\bar{x}^2\bar{y}^2 - 2r^2\bar{x}^2 - 2r^2\bar{y}^2 + r^2 + z^2}} \tag{3.10}$$

$$\vec{B} = \begin{bmatrix} 2\bar{x}^3 + 2\bar{x}\bar{y}^2 - 2r^2\bar{x} \\ 2\bar{y}^3 + 2\bar{x}^2\bar{y} - 2r^2\bar{y} \\ z \end{bmatrix} \tag{3.11}$$

and

$$\bar{x} = x - x_c \tag{3.12}$$

$$\bar{y} = y - y_c \tag{3.13}$$

Evaluating equation 3.9 results in a vector field that converges to a circular path. Circulation is calculated from the cross product of each implicit surface function's gradient, which simplifies to

$$\overrightarrow{V}_{circ} = \begin{bmatrix} 2(y - y_c) \\ -2(x - x_c) \\ 0 \end{bmatrix} \tag{3.14}$$

The guidance $\overrightarrow{V}_o$ described by intersecting a cylinder and a plane can be shown in Figure 3.2. Note that repulsion from the path that lies at the intersection is achieved by a negative convergence weight, $G < 0$.

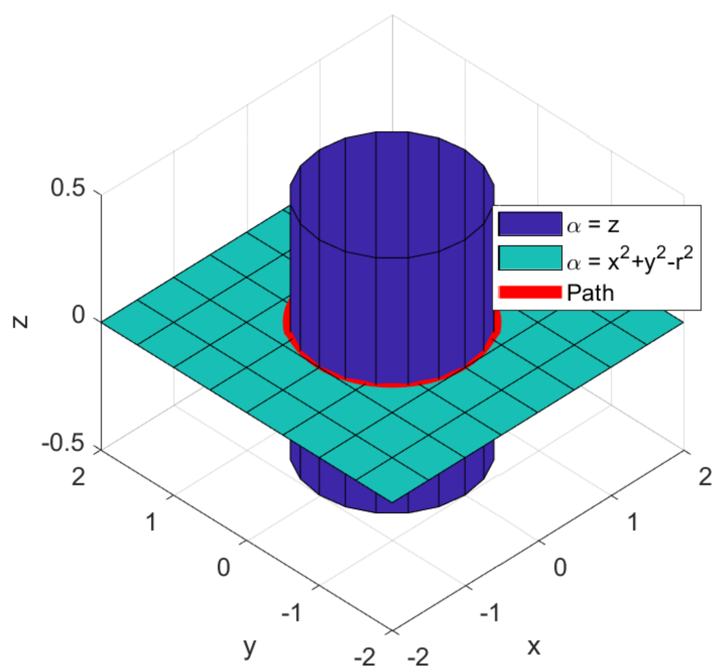


Figure 3.2: Intersection of a cylinder and plane defined by implicit surface functions

Obstacle fields should only act locally on a UAV guidance which is accomplished by applying a decay function for a field of radius R. The decay strength P is determined in 3.15, where d is the euclidean distance, or range, between the UAV and the center of the obstacle, shown in Equation 3.16. At a distance $d > R$ the decay strength P is effectively zero, having virtual no influence on the total guidance. At a distance $d \leq R$, the field strength is bounded between $[0, 2]$. The selection of the decay function P to be bounded as such is so that the obstacle field $\overrightarrow{V}_o$ eventually overpowers the path field $\overrightarrow{V}_{||P||}$.

$$P = -\tanh\left(\frac{2\pi d}{R} - \pi\right) + 1 \tag{3.15}$$

$$d = \sqrt{\bar{x}^2 + \bar{y}^2} \tag{3.16}$$

Prior to applying the decay function P the obstacle field $\overrightarrow{V}_o$ is normalized. Forcing the obstacle guidance $\overrightarrow{V}_{||o||}$ to have unity magnitude, bounds the decayed guidance magnitude $P\overrightarrow{V}_{||o||}$ to the interval $[0, 2]$ which allows a prediction of singularity location based on the range from obstacle center, d.

$$\overrightarrow{V}_{||o||} = \frac{\overrightarrow{V}_o}{||\overrightarrow{V}_o||} \tag{3.17}$$

3.2.3 Summed Guidance and Singularity Definition

The total summed guidance $\overrightarrow{V}_g$ defined in Equation 3.6 can now be calculated and visualized, Figure 3.3. In general, the GVF guidance is more easily visualized if the final guidance is normalized yet again, however, plotting the guidance $\overrightarrow{V}_g$ demonstrates regions where the fields oppose each other and decrease the total fields magnitude. For a strictly repulsive field that is centered on a straight path, regions of both constructive and destructive summation occurs. Note the vectors on the positive x-axis increasing in length as the two fields come together. Conversely, vectors in the negative x-axis show decreasing length and in some areas disappearing entirely.

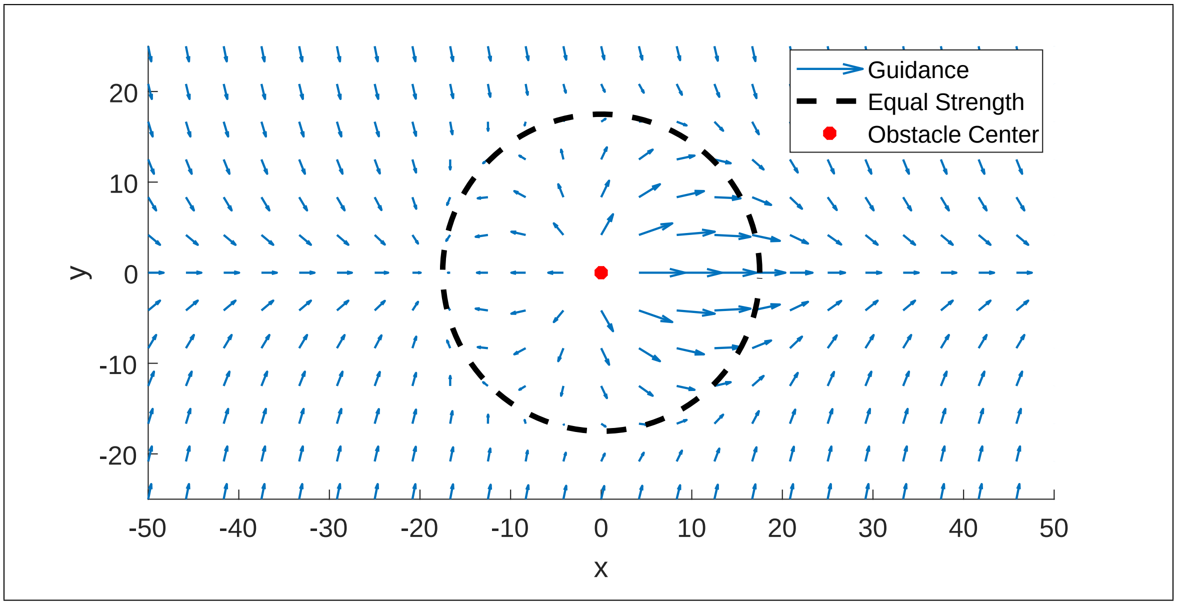


Figure 3.3: Summed fields without total normalization $\overrightarrow{V}_g$

Observing the regions around disappearing vectors it is shown that vectors appear to converge to a singular point from all direction, possibly indicating a trap situation. Visualizing the vector magnitudes of the scenario presented above can be accomplished with a summed field heat map, shown in Figure 3.4, which shows the regions of decreasing magnitude more clearly.

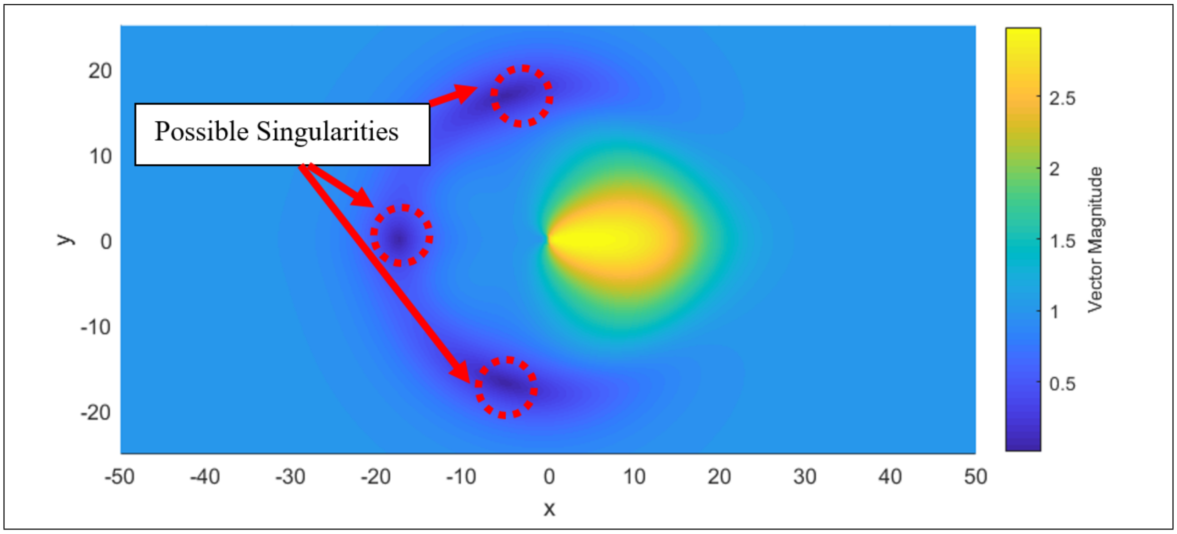


Figure 3.4: Summed Fields Without Total Normalization

If singularities indicate possible trap situations, it is important to detect and avoid them. Singularities in the vector field are defined as a region in the GVF space where the vector has zero magnitude, shown in Equation 3.18.

$$||\overrightarrow{V}_g|| = 0 \tag{3.18}$$

By extension, singularities are a result of a zero vector, shown in Equation 3.19 and Equation 3.20.

$$\overrightarrow{0} = \overrightarrow{V}_{||P||} + P\overrightarrow{V}_{||O||} \tag{3.19}$$

$$\overrightarrow{V}_{||P||} = -P\overrightarrow{V}_{||O||} \tag{3.20}$$

Vectors $\overrightarrow{V}_{||P||}$ and $\overrightarrow{V}_{||O||}$ are normalized, meaning that their magnitudes are of equal length $||\overrightarrow{V}_{||P||}|| = ||\overrightarrow{V}_{||O||}||$. For the condition shown in Equation 3.20 to be true for an obstacle field with a negative convergence weight $G = -1$, the decay function P must be unity.

Setting Equation 3.15 equal to 1, the distance at which the fields have equal strength, and therefore possible singularity locations, is determined to be that shown in Equation 3.21

$$d = \frac{R}{2} \tag{3.21}$$

Searching the GVF for locations that satisfy Equation 3.18 can be accomplished numerically, however a good initial condition is necessary. Initial conditions placed at the radius of equal strength, defined by Equation 3.21, on the side of the obstacle where deconstruction summation occurs the singularities can be found. Solving for the singularity locations with initial conditions placed evenly on the right hand side of the obstacle of the above scenario locates three singularities located on the radius of equation strength, shown in Figure 3.5.

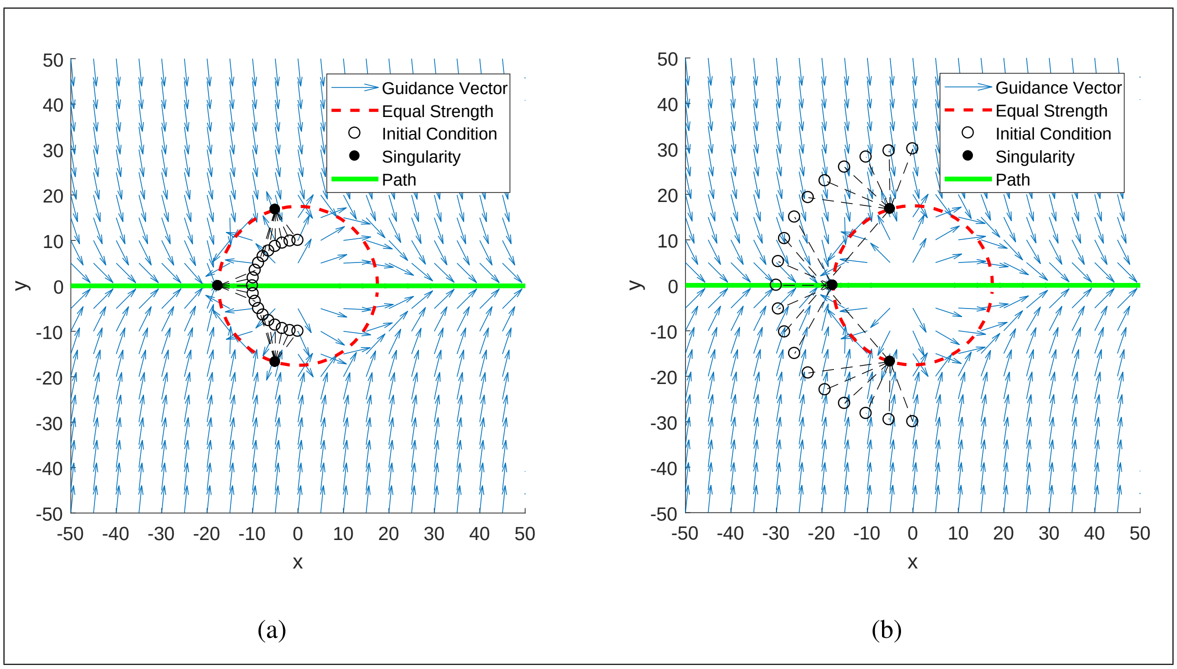


Figure 3.5: GVF converging and circulating circular path

By definition singularities are caused by the two vector fields directly opposing each other, described in Equation 3.20. Adding circulation to the repulsive GVF may prevent the summed vectors from canceling out completely. To demonstrate this, the above scenario is repeated with equal magnitude convergence and circulation components. Note that by adding circulation to the obstacle vector field, it prevents singularities from forming along the route the UAV is likely to take. Additionally, adding circulation provides a deterministic guidance on how to circumnavigate an obstacle. In literature, obstacles directed the UAV away and relied on the path following field for direction. Generate (or find) figure with circulation added to show singularities removed from path

3.3 Phase II: Optimization of Obstacle Field

The objective of Phase II is to determine a combination of GVF circulation and decay radius for an optimized circular obstacle avoidance. A demonstration of a UAV modeled as a Dubin's vehicle converging and following a straight path for various target path circulations is provided. A circular obstacle represented by a strictly repulsive GVF will be added to the path and the avoidance observed. Obstacles radius and their associated GVF decay radius will now be represented in terms of the UAVs turning radius for convenience. Next, a path deviation cost function is described and is to be minimized to provide an optimized GVF obstacle avoidance guidance. It is shown how cost is effected by modifying the decay radius and circulation for the worst-case obstacle avoidance scenario presented. Lastly, a method for solving for circulation and decay radius numerically is presented.

3.3.1 Vehicle and Obstacle Definition

As discussed in literature and Phase I, GVF can be used to guide a UAV to get on and follow a path. An example of a UAV traveling at constant speed $u = 20m/s$ and fixed turn rate $\dot{\theta} = 20deg/s$ converging and circulating a straight path using the GVF guidance described in Phase I is shown in Figure 3.6. Starting at an initial position of $(-45, 20)m$ and heading θ of 45°, the UAV is shown converging and following a straight GVF path with weights $G = 1, H = 5$.

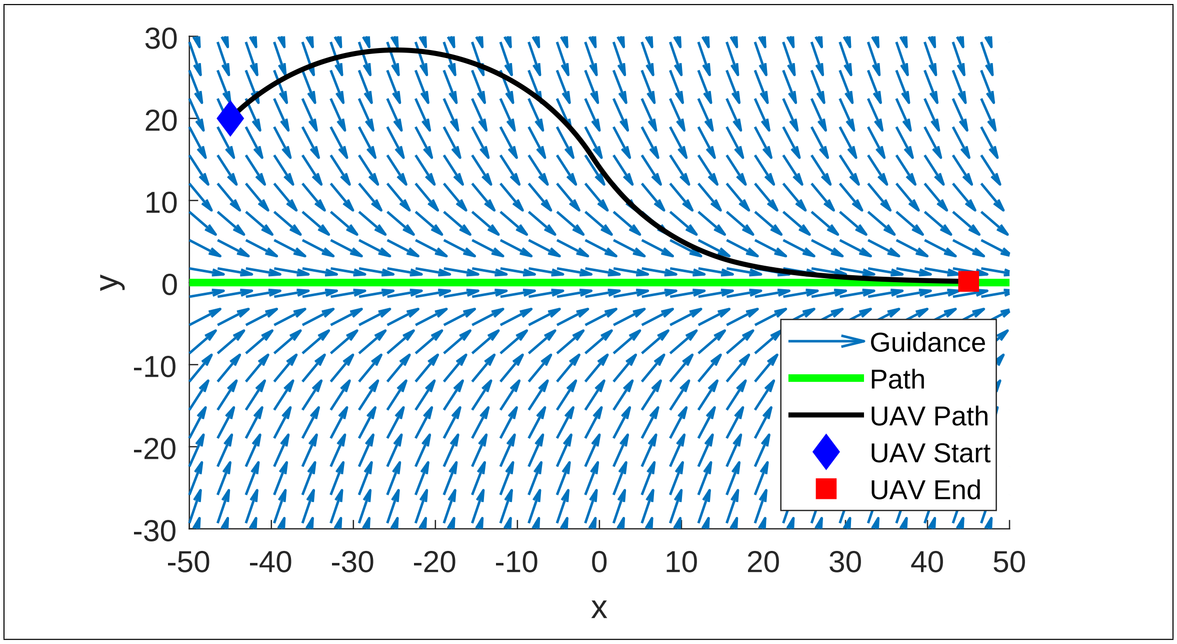


Figure 3.6: Fixed Wing converging and following a path

The effect of modifying G and H has not been well documented in literature when the GVF is normalized for a heading guidance. To determine an appropriate relationship between circulation and UAV velocity, various path circulation weights for a fixed UAV velocity were conducted and the lateral error with respect to time recorded. As discussed in Phase I, increased circulation is expected to result in the vector field transitioning to path following more quickly. A low circulation weight allows convergence to dominate closer to the path. The UAVs route for converging and following a path with $G = 1$ and multiple path circulation weights H is shown in Figure 3.7. As expected, a low circulation results in a UAV route that quickly approaches the path, however begins to oscillate and has a larger stead-state error compared to larger circulation fields. The largest circulation field $H = 50$, does not oscillate, but takes considerably longer to converge. Circulation approximately the same value as the UAV's velocity results in a compromise between fast convergence and no oscillation. The lateral error from the path is shown in Figure 3.8.

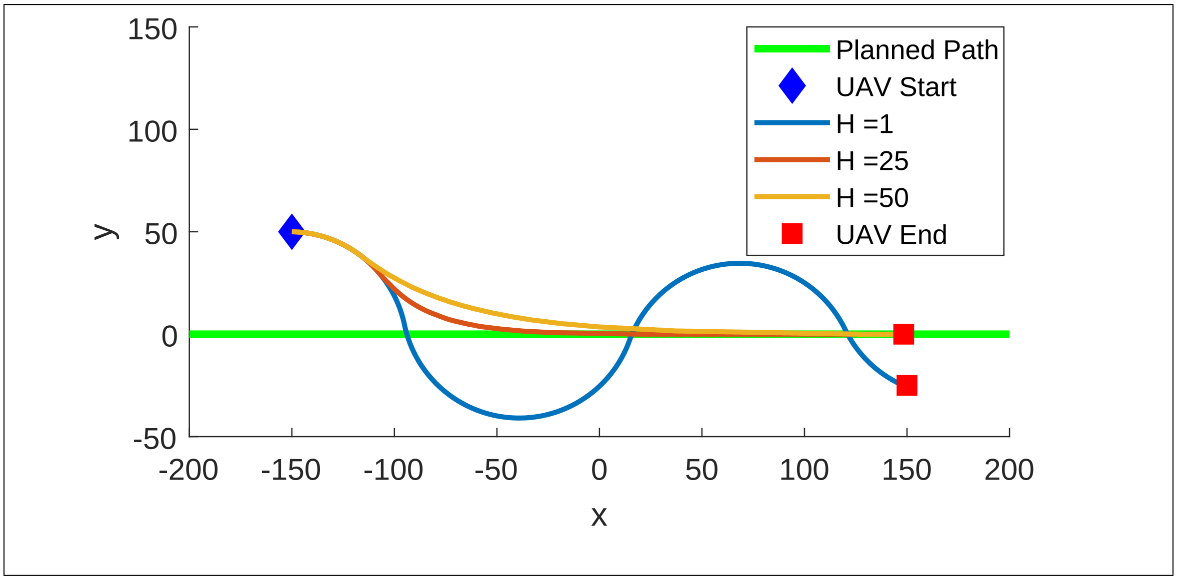


Figure 3.7: Fixed Wing converging and following a path

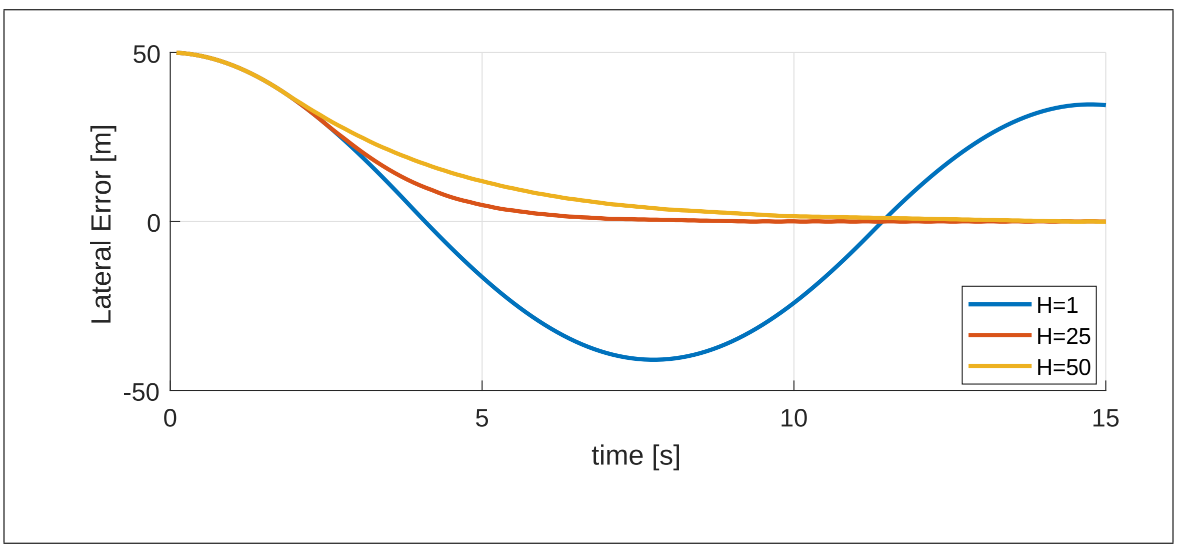


Figure 3.8: Lateral error for fixed wing guided by GVF guidance of multiple circulations

From the above simulation it is observed that a circulation near the UAVs velocity provides a flight route that is in-between the high oscillation and high overshoot of a circulation $H = 1$ while not having a longer settling time with high circulation $H = 50$. For future simulations the circulation of the path following guidance will be assumed to be equal to that of the UAVs velocity, $H = u$.

As the UAV travels the path using GVF guidance, an obstacle may be encountered that, without intervention, the UAV will collide with. Obstacles along the pre-planned path are described using two parameters, the obstacle radius r_o and the lateral distance from the path y_o. Representing the obstacle's position with respect to the path is useful because it allows for a single optimized GVF solution to be applicable for multiple path angles δ.

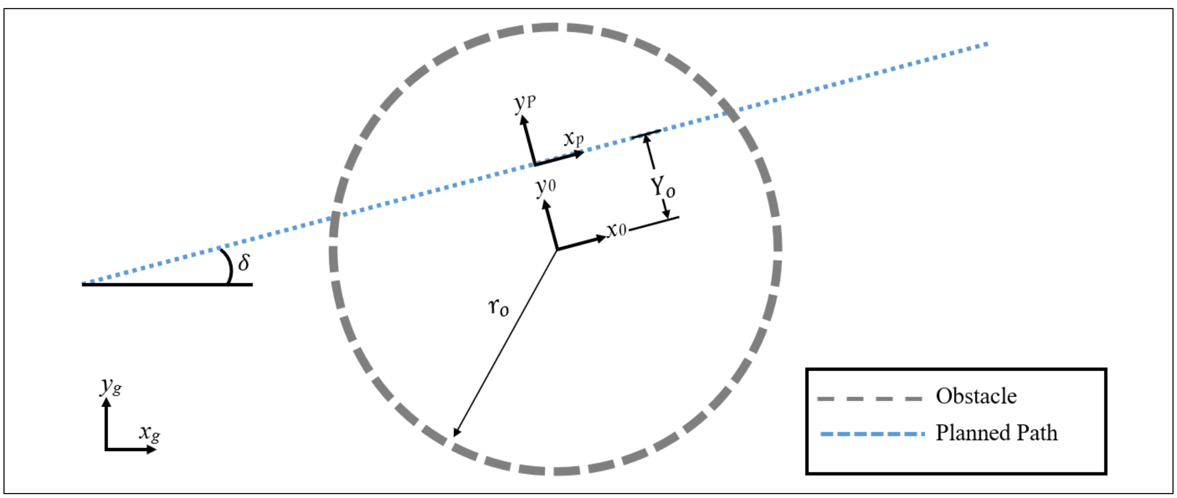


Figure 3.9: Circular obstacle along planned path

It is assumed here that the radius of the obstacle is no smaller than the turn radius of the UAV θ_r, which is calculated in Equation 3.22. It is convenient to represent the obstacle's radius as m multiples of the UAV's turning radius, shown in Equation 3.23.

$$\theta_r = \frac{u}{\dot{\theta}} \tag{3.22}$$

$$r_o = m\theta_r \tag{3.23}$$

The repulsive vector field's decay radius R is defined in k multiples of the obstacle's radius, shown in Equation 3.24.

$$R = kr_o \tag{3.24}$$

A cost function can be used to measure the deviation from a planned path while avoiding an obstacle with GVF in Equation 3.25. The deviation, y, is the lateral distance of the UAV to the planned path, R is the obstacle radius.

$$\gamma = \frac{1}{r_o}\int_0^{tf} ydt \tag{3.25}$$

Strictly repulsive GVF obstacle fields can provide avoidance, however may cause excess deviation from the pre-planned path, cause unnecessary turns, and flight routes near or through guidance singularities. A UAV traveling at a speed of $10m/s$ and a turn rate of $20deg/s$ is given a summed path following and obstacle avoidance guidance $\overrightarrow{V}_g$, defined in Equation 3.6 in Phase I. An obstacle centered on the path $y_o = 0$ and radius $r_0 = \theta_r$ is to be avoided. The decay radius multiplier k was increased manually over several simulations until avoidance was achieved. The flight path of the UAV flying with summed guidance is shown in Figure 3.10. The UAV experiences excessive path deviation and takes considerable time to settle back to the planned path. Additionally, the guidance directs the UAV to pass directly through a singularity and passes near an additional singularity towards the top of the obstacle. The cost, calculated from Equation 3.25, for avoidance using the strictly repulsive guidance for head on collision scenario is $\gamma = 25$.

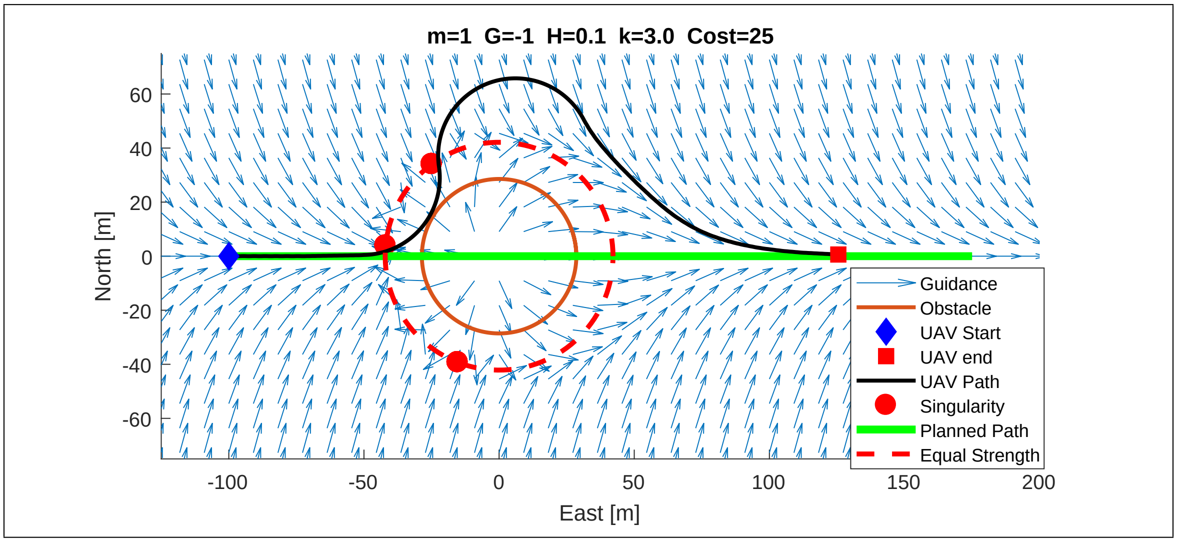


Figure 3.10: UAV encountering a circular obstacle centered on pre-planned path, no circulation

Adding circulation to the obstacle guidance $\overrightarrow{V}_o$, as described in Phase I, may remove singularities from the UAVs route. Additionally, circulation adds deterministic information on how to circumnavigate an obstacle whereas strictly repulsive fields rely on the path following field to determine circumnavigation direction. Equal magnitudes convergence G_o and circulation H_o removes the guidance singularities from the UAV's path, adds a more gentle transition between fields, and guides the UAV back to the planned path more quickly, shown in Figure3.11. The cost of the head on collision avoidance using GVF guidance with obstacle circulation results in a cost of $\gamma = 7$.

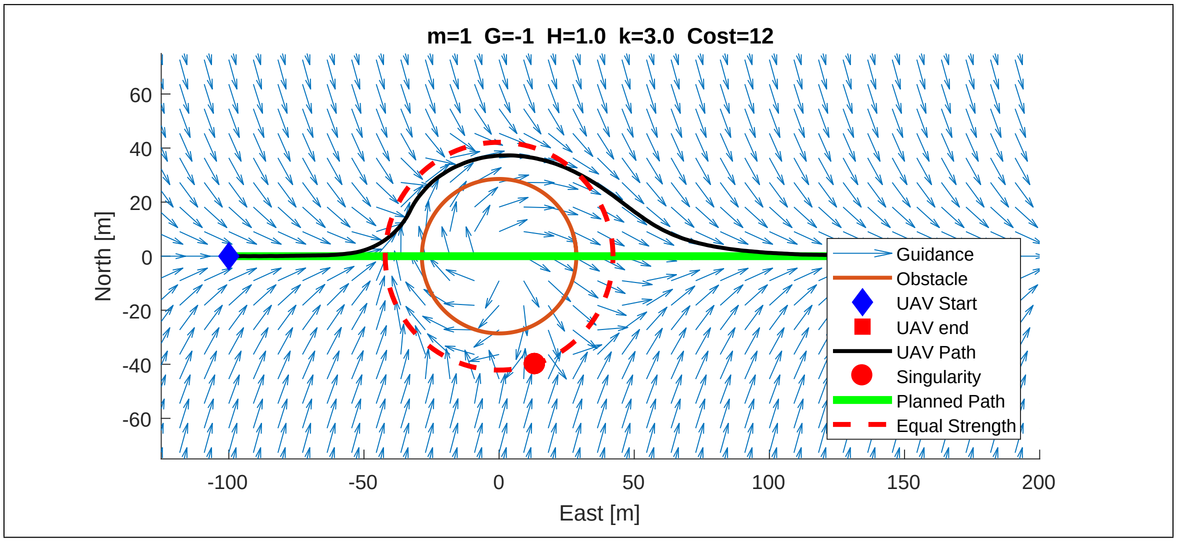


Figure 3.11: UAV encountering a circular obstacle centered on pre-planned path with circulation

Adding equal parts G and H produce guidance with a lower cost compared to strictly repulsive, however, k and H should be selected such that the cost is minimized. One method for selecting what these field parameters should be is to evaluate a large range of parameters and observe the cost for avoidance with each combination. The cost function shown in Equation 3.25 only penalized the UAV for path deviation, however should be modified to also be penalized for violating the obstacle space. The new cost function $\bar{\gamma}$ adds an additional piecewise function which increases the cost if the UAV is inside or on the obstacle's edge, shown in Equation 3.26

$$\bar{\gamma} = \frac{1}{r_o} \int_0^{tf} ydt + j(x, y) \tag{3.26}$$

$$j(x, y) = \begin{cases} 100dt & \sqrt{(x - xc)^2 + (y - yc)^2} \leq r_O \\ 0 & \sqrt{(x - xc)^2 + (y - yc)^2} > r_O \end{cases} \tag{3.27}$$

Using the above cost function $\bar{\gamma}$ in the same scenario presented in Figure 3.10, several simulations were conducted with identical UAV parameters and obstacle radius r_o

with varying k and H_o values. The cost of each simulation is shown in the heatmap in Figure 3.12.

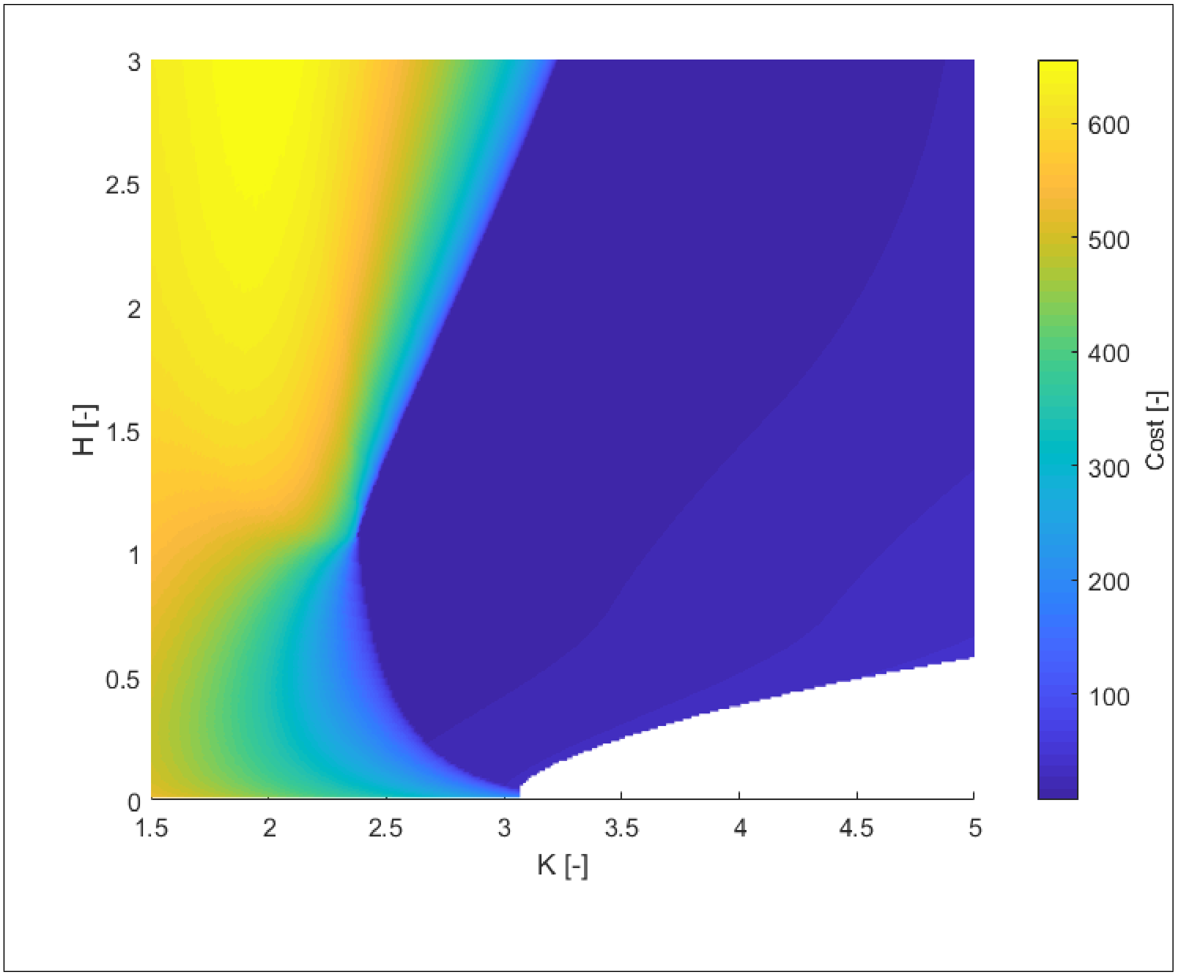


Figure 3.12: Heatmap of cost as function of k and H_o

Finding the minimum cost of the above heatmap and looking up the corresponding k and H_o values at which the minimum occurs can be used to provide an optimized avoidance. Solving the desired parameters for an optimal guidance using this search method for the general case is computationally expensive and can take many hundreds of simulations to evaluate the desired space. The problem becomes one of minimization of the cost function and to find a solution numerically without solving for a large range of parameters, shown in Equation 3.28.

$$\underset{H,k}{\text{minimize}} \quad \frac{1}{R}\int_0^{tf} y dt + j(x,y) \tag{3.28}$$

A numerical solver which attempts to minimize a given cost function can be found in MATLAB called $fmincon()$. The minimizer operates on the following principle. Provided an initial condition array X_I, minimize the function $\bar{y}$ by observing the change in $\bar{y}$ within certain bounds. Using the minimizer method a solution to the above problem of a UAV following a path with a head-on collision scenario with an obstacle was found in 5.2 seconds compared to [insert time for finding solution with heat map]. The reduction in cost as the minimizer runs is shown in Figure 3.13.

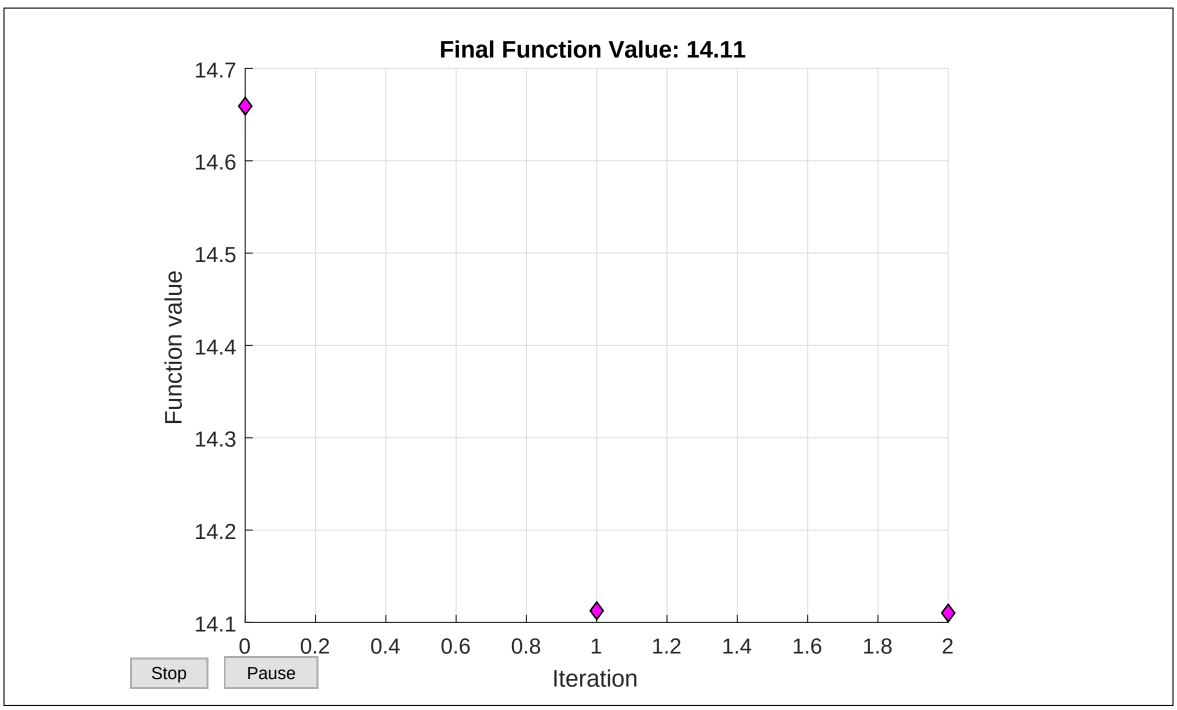
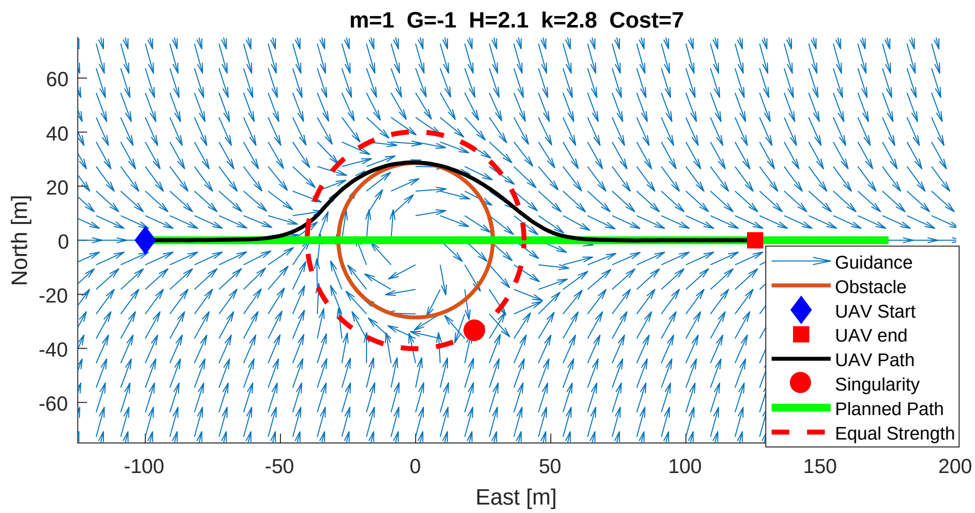


Figure 3.13: Cost function reduction in fmincon()

The optimizer found a solution combination of $k = 2.7$ and $H_o = 1.8$ and results in a path with cost $\bar{\gamma} = 14$. The path can be seen below in Figure 3.14. Discuss the percent difference from all the methods, possibly make a table discussing the computation time



mm

Figure 3.14: UAV path from optimized GVF

The optimized GVF guidance was shown to provide an improved guidance with a reduced path deviation compared to standard GVF guidance. Determining how GVF compares against other methods such as VFF, waypoint, and the optimal path around an obstacle is now discussed.

3.3.2 Optimal Avoidance Route for Straight Path

A geometrically optimal route around a circular obstacle can be used to compare the performance of avoidance algorithms. The path for avoiding a circular obstacle while minimizing path deviation can be accomplished with three circular arc turns. The first and third arc utilize the UAVs minimum turning radius, θ_r, calculated in Equation 3.29. The start of the first minimum radius turn begins when the UAV's horizontal position x reaches

$\tilde{x}$ from the path frame origin. At a horizontal position $-\hat{x}$ the UAV turns with a radius of the obstacle r_o and exits when the UAVs horizontal position reaches $\hat{x}$.

$$\theta_r = \frac{u}{\dot{\theta}} \tag{3.29}$$

The horizontal points $\tilde{x}$ and $\hat{x}$ are shown in Equations 3.30 and 3.31 respectively.

$$\widetilde{x} = -\sqrt{(\theta_r + R)^2 - (\theta_r - Y_o)^2} \tag{3.30}$$

$$\hat{x} = \frac{R\sqrt{(r + R)^2 - (\theta_r - Y_o)^2}}{R + \theta_r} \tag{3.31}$$

The avoidance path for navigating around a circular obstacle with maximum coverage of a sensor line is defined in Equation 3.32 and shown in Figure 3.15.

$$y(x) = \begin{cases} \widetilde{y} - \sqrt{\theta_r^2 - (x - \widetilde{x})^2} & x < -\hat{x} \\ Y_o + \sqrt{R^2 - x^2} & -\hat{x} \leq x \geq \hat{x} \\ \widetilde{y} - \sqrt{\theta_r^2 - (x + \widetilde{x})^2} & x > -\hat{x} \end{cases} \tag{3.32}$$

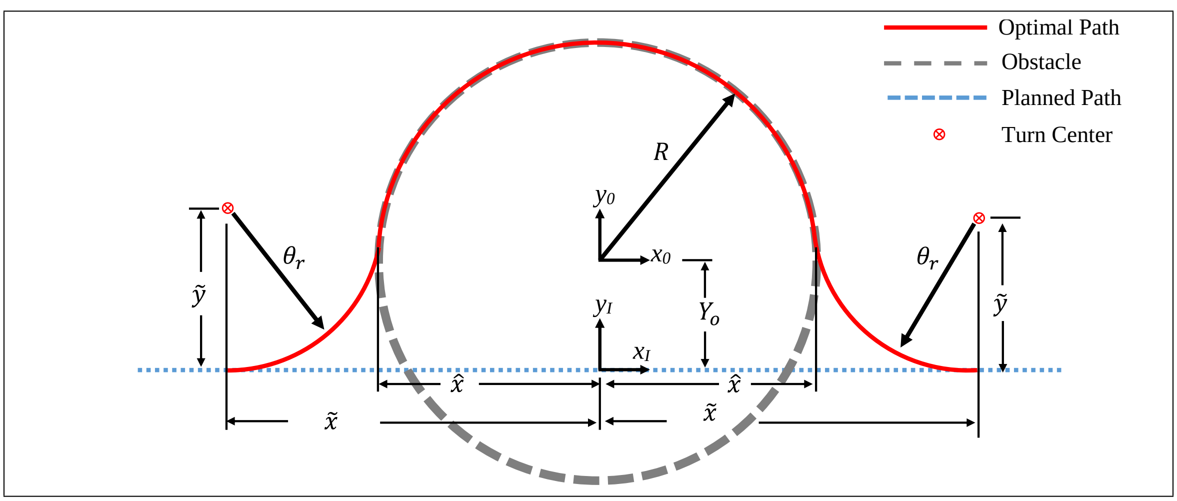


Figure 3.15: Optimal Kinematic Path Around Circular Obstacle

The avoidance path represents the optimal path around a circular obstacle and would be used to generate waypoints for waypoint guidance.

3.3.3 Optimal Avoidance Path and Waypoints

The number of waypoints that divert around an obstacle effects how closely the UAV tracks the outside of the obstacle and how much of the original path can be traveled. Few obstacle diversion waypoints leads to excess path deviation while increasing the number of diversion waypoints reduces path deviation, however has diminishing returns. The cost function in Equation 3.25 is used below to demonstrate how increasing the number of waypoints decreases the cost function, however approaches an asymptote around six waypoints.

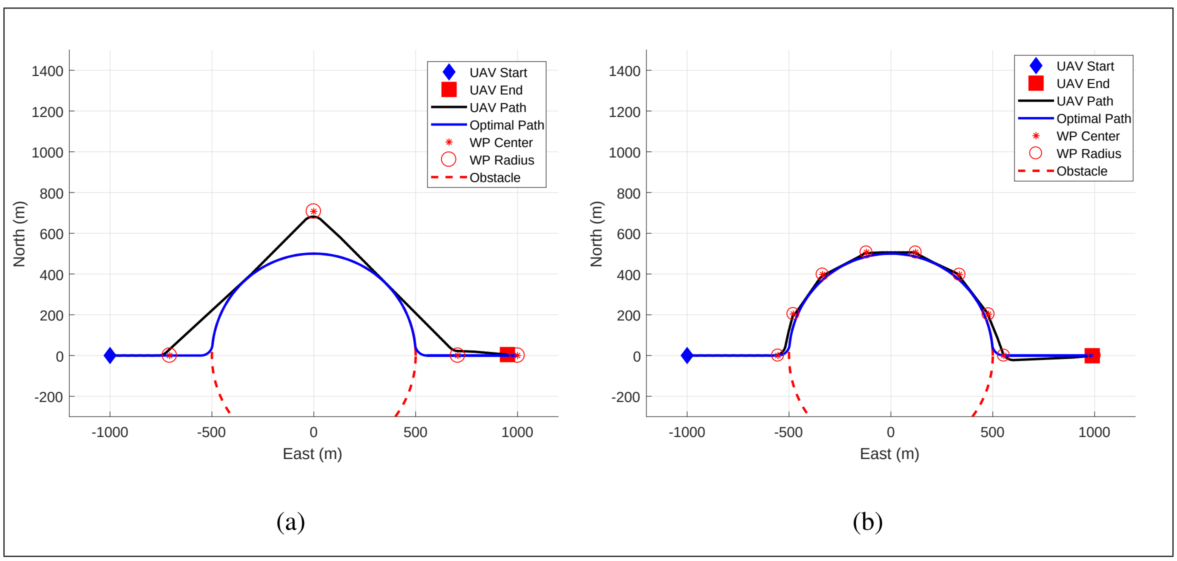


Figure 3.16: Obstacle Diversion Waypoints

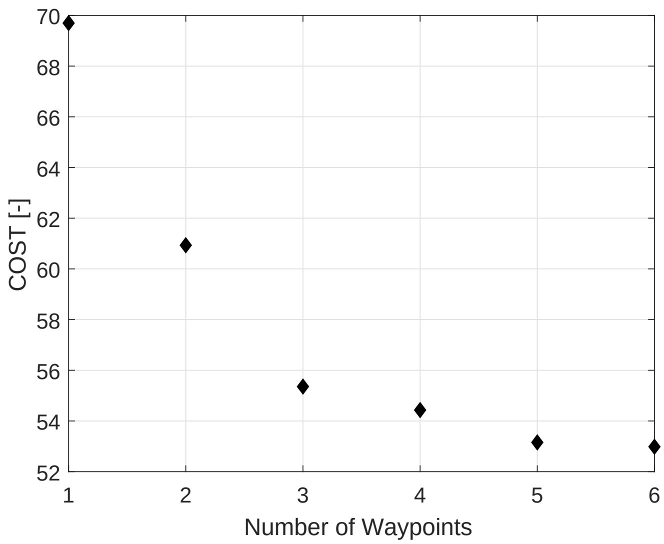


Figure 3.17: Cost impact versus number of waypoints

3.3.4 Worst Case Avoidance Scenario

A worst case avoidance scenario will be used to compare the optimized GVF with waypoint, VFF, and the optimal path with respect to the path deviation cost function. A circular obstacle centered on the path, $y_o = 0$, requires a deviation from the path of at least 50% of the obstacles radius. A fixed wing UAV at an initial position $(-400, 0)$ and heading $\theta = 0°$ follows the straight path connecting the points $(-400, 0)$ and $(400, 0)$ respectively. Traveling at a constant speed $u = 25m/s$ and with a fixed turn rate of $\dot{\theta} = 20deg/s$ the UAV must avoid an obstacle with radius $2\theta_r$ located at the origin $(0, 0)$. The VFF guidance from [26] is used with an obstacle window radius of $\theta_r + r_o$, a cell repulsion $Fr = -3$, attraction force $F_t = 0.8$, range exponent $n = 2$, and a goal located at $(700, 0)$. For LOS waypoint guidance, 7 waypoints with a small waypoint radius of $10m$ was chosen. Each diversion waypoint added drives the guidance closer to optimal, however has diminishing returns past $6 - 7$ waypoints. GVF guidance with a circular repulsive field was assigned a convergence weight $G = -1$ and circulation and decay radius coefficient k were determined by evaluating the cost function in Equation 3.25 with initial conditions $k_i = 2$ and $H = 2$. The GVF solution was bounded such that $2 \leq k \leq 4$ and $1 \leq H \leq 6$. Minimizing the cost function resulted in a decay radius coefficient $k = 2.78$ and a circulation value $H = 1.88$. The Dubin's paths for the three guidance methods discussed is shown in Figure 3.18.

VFF results in a UAV route that has excess deviation from the planned path with excessive turns. Waypoint guidance returns to the path more quickly than VFF, however deviates from the planned path farther then necessary. GVF leaves the path before waypoint guidance and tracks the outside of the obstacle closely and then quickly converges back to the pre-planned path. The cost of each method, defined in Equation 3.25, is displayed in the bar plot shown in Figure 3.19.

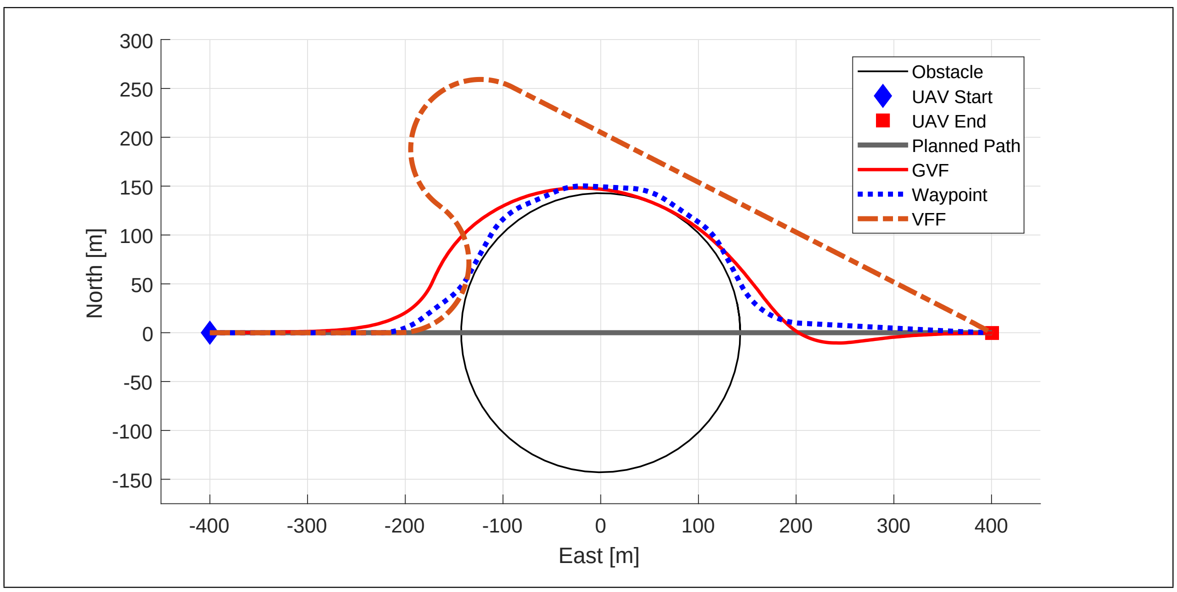


Figure 3.18: Path of UAV guided by guidance methods

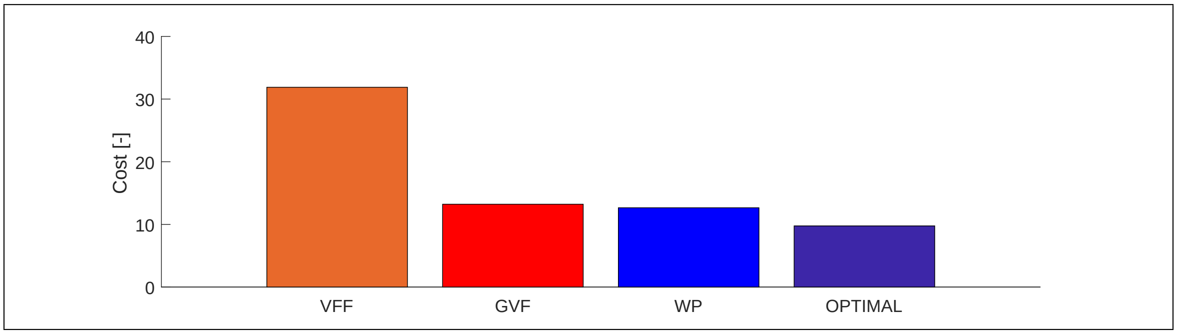


Figure 3.19: Cost performance for various UAV guidance methods

3.4 Phase III

The objective of Phase III is to demonstrate the optimized gradient vector field guidance presented in Phase II on multirotor UAV flying with fixed wing turn-rate constraints. A description of the experimental setup will be given for both the hardware and software that was implemented to achieve GVF guidance. An overview of the validation of a conversion to python from MATLAB is discussed. Lastly a description of an avoidance scenario is presented.

3.4.1 Experimental Overview

All of the scenarios discussed using GVF guidance have involved simulating a fixed wing UAV modeled as a Dubin's vehicle. To demonstrate the GVF on actual flight hardware, an indoor quadcopter, Figure 3.20, will be used in place of a fixed wing UAV. There are several reasons why using an indoor quadcopter is advantageous for experimental flight tests, such as new guidance systems. First, finding an airspace to safely test the guidance system with an adequate clearing for takeoff and landing can be difficult. Many environmental complications such as high winds, precipitation, and low visibility could delay or prevent flight tests all together. Testing the GVF guidance method on an indoor quadcopter remove these complications. Additionally, the Dubin's turn rate constrains can be applied to limit the maneuverability of the quadcopter so that it behaves similar to that of a fixed wing UAV. Flying indoors also allows for more repeatable experimentation, environmental control, and use of high speed motion capture systems to provide position information to guidance and control systems.

Figure 3.20: Micro quadcopter Crazyflie 2.0 by bitcraze

The micro quadcopter, crazyflie 2.0, designed by Bitcraze was selected as the experimental flight vehicle due to its low cost and the ability to send the vehicle roll rate, pitch rate, yaw, and thrust commands directly over radio. The control messages can be sent through the object oriented and scripted language Python, a language with syntax very similar to MATLAB. The UAV was viewed by 8 vicon vero cameras detect infrared light reflected by small markers placed on the vehicle. Video captured by the cameras at 100Hz is sent to a PC with a software package that estimates the pose of the vehicle and sends that information over a local area network (LAN) to a ground station PC where guidance and control calculations are made. The command messages are then sent over radio to the crazyflie where an on-board controller accepts the commands and outputs the necessary motor output to achieve the commands. A high level overview of the experimental framework described is shown in Figure 3.21. Each component of the framework will be discussed in more detail in the proceeding sections.

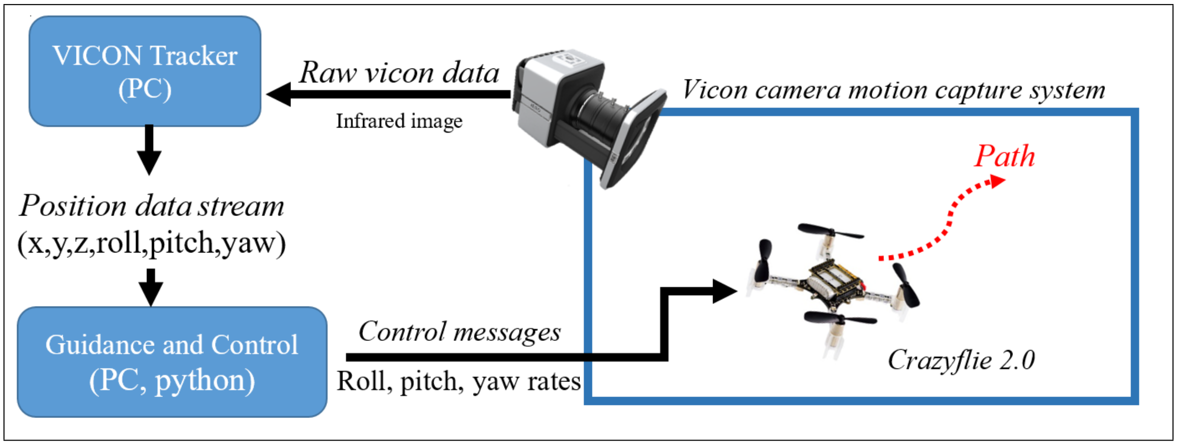


Figure 3.21: Indoor quadcopter flight experimental layout

3.4.2 Crazyflie 2.0

The crazieflie 2.0 is a lightweight micro-quadcopter UAV weighing in at 27 grams and has an approximate payload capacity of 10 grams. An on-board flight controller maintains vehicle stability by estimating it's attitude and making corrections to the four brushed motors. One of the conveniences of the crazyflie 2.0 is the ability to send roll rate, pitch rate, yaw, and thrust over radio in order to control the UAV directly. The software package, cfClient, interfaces with the crazyflie to accept and transmits these control messages over radio.

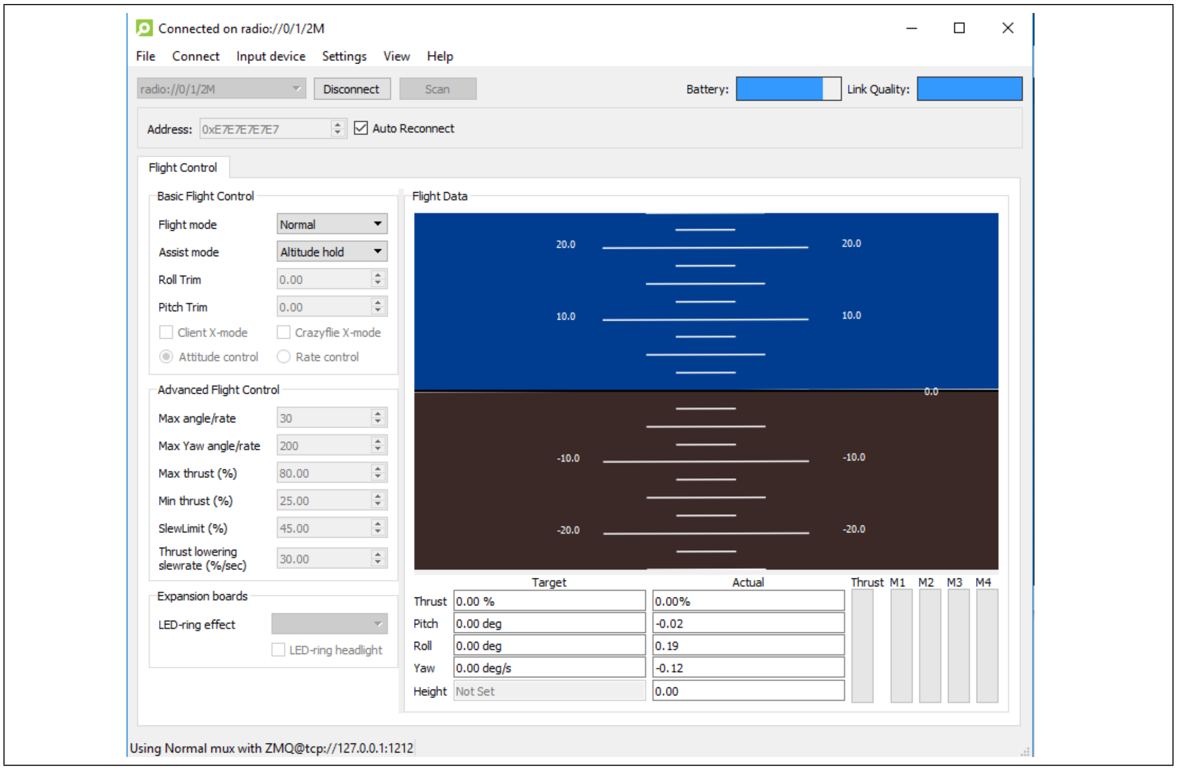


Figure 3.22: Crazieflie client radio communication software

The guidance and control software that calculates these command messages was hosted on a remote machine and not on-board the UAV, primarily for the convenience of fast development. The guidance and control was written in Python, which is highly portable and could easily be implemented on-board a UAV.

3.4.3 Python Guidance and Control Ground Station

The guidance and control framework developed for experimentation resembles that described in Figure 2.3. A navigation script taps into a stream of data provided by the Vicon tracker software and distributes that data to guidance and control algorithms. Position (x, y, z) and yaw ϕ are provided to the control algorithm which consists of four PID controllers. The UAV was set to fly at constant altitude for all simulations and experimentation, therefore only planar position (x, y) were provided to the guidance system.

Heading guidance from the optimized GVF was converted to a carrot located at a position (xc, yc) is sent to the control system to be used as set-points. Control messages are relayed to a radio interface software, cfClient, which communicates the control messages with the crazyflie. An overview of the described system is shown in Figure 3.23.

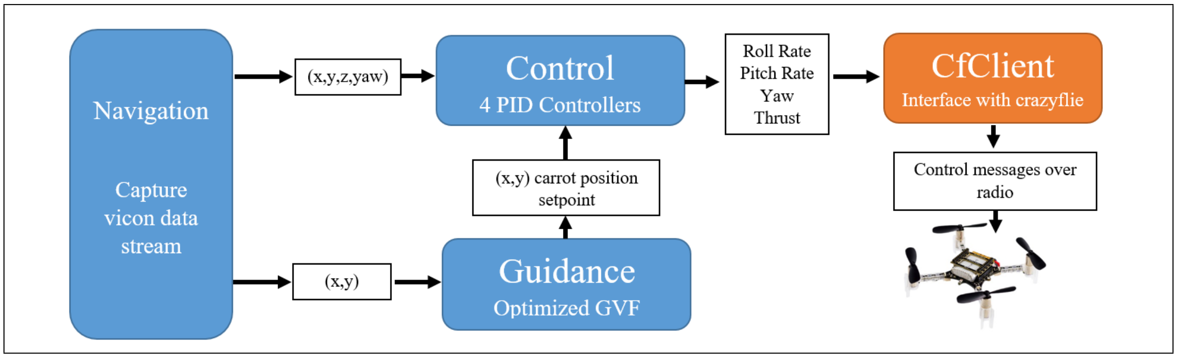


Figure 3.23: Crazyflie Guidance and Control Software Framework

The control algorithm consists of four PID controllers which are used to calculate roll rate, pitch rate, yaw, and thrust to drive the UAV to a desired setpoint. The error is measured by subtracting the measured state (x, y, z, yaw) from the desired setpoint. The block diagram of the PID controller is shown in Figure 3.24. Gains P, I, and D for each controller is tabulated in Table 3.1 below.

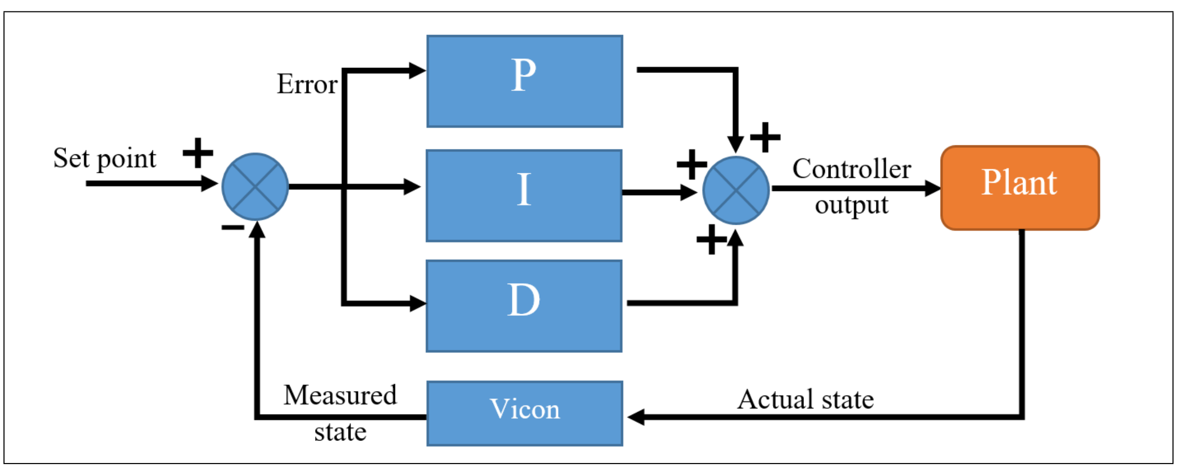


Figure 3.24: Crazyflie Guidance and Control Software Framework

Table 3.1: Tuned PID gains for roll, pitch, yaw, and thrust controller

Control Parameter	P	I	D
Roll Rate	29	2.5	19
Pitch Rate	29	2.5	19
Yaw	80	50	30
Thrust	100	90	70

The optimized GVF guidance developed in MATLAB in Phases I and II was programmed into Python and compared under several scenarios to ensure that the methods were identical. First, a path following GVF was calculated in Python for a straight line and results overlaid with an identical scenario in MATLAB. The quiver plots show guidance calculated by MATLAB aligning with the guidance calculated in Python shown in Figure 3.25.

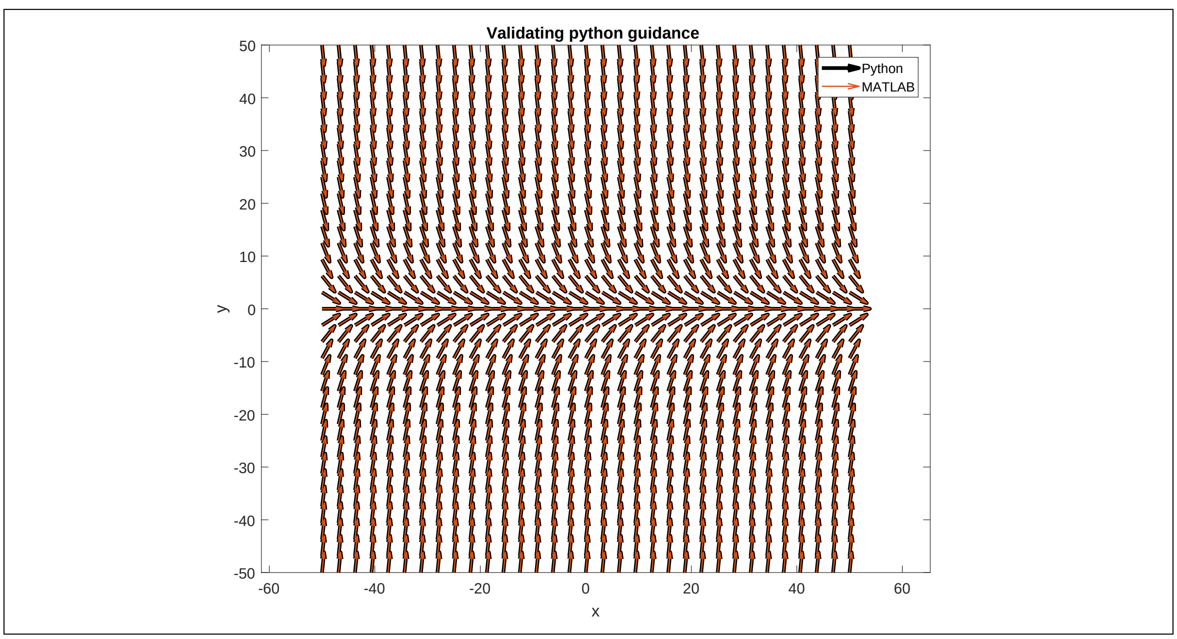


Figure 3.25: Validation of Python straight path guidance overlaid with MATLAB

Validating for an avoidance field with equal parts circulation and repulsion are shown in Figure 3.26

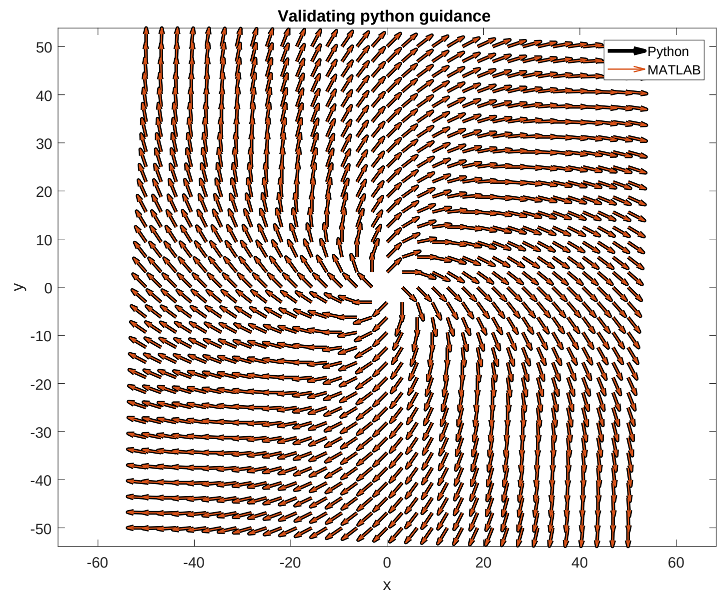


Figure 3.26: Validation of Python obstacle guidance overlaid with MATLAB

Avoidance field with the decay applied in Figure 3.27

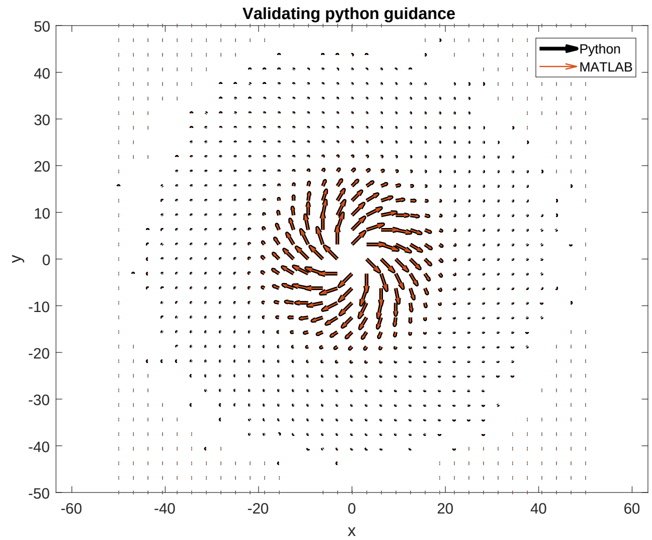


Figure 3.27: Validation of Python obstacle decay guidance overlaid with MATLAB

Summed path following and obstacle avoidance guidance with an obstacle field with no circulation is shown in Figure 3.28.

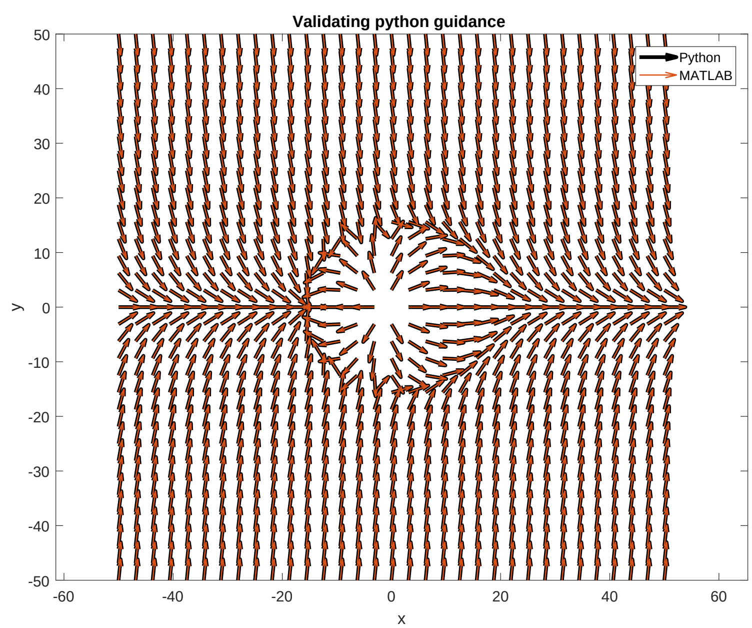


Figure 3.28: Validation of Python summed guidance overlaid with MATLAB

Lastly, the Python guidance was simulated with an identical Dubins vehicle as MATLAB and compared in Figure 3.29

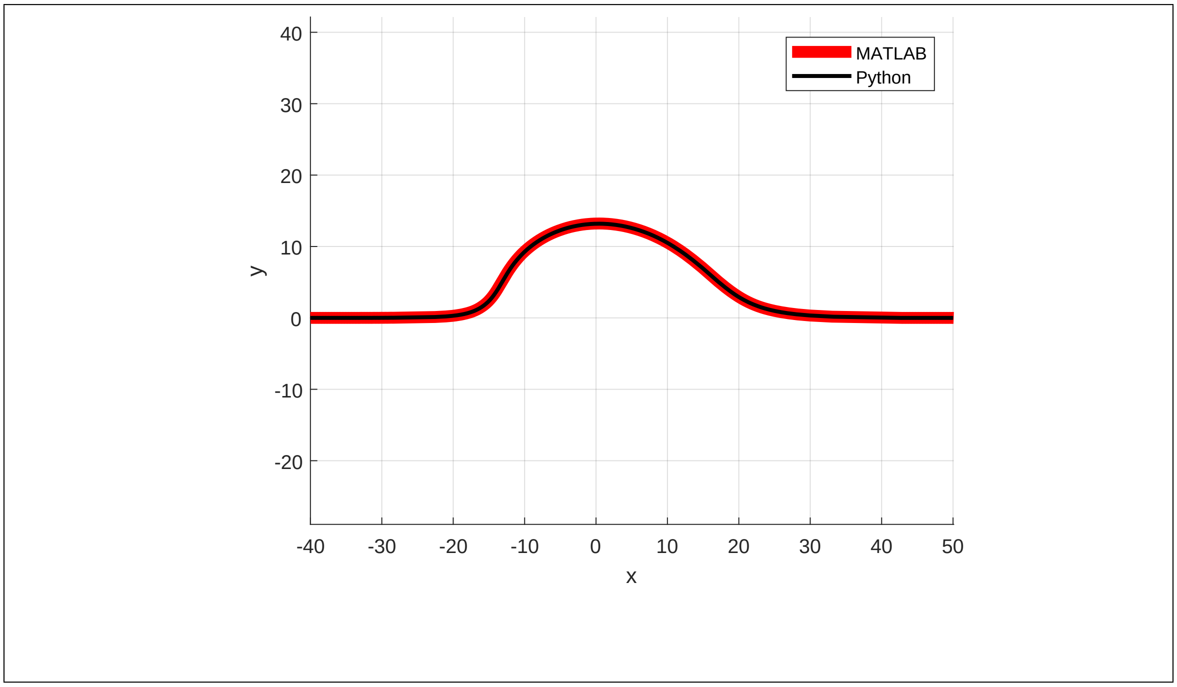


Figure 3.29: Validation of Python Dubins UAV route overlaid with MATLAB

3.5 Summary of Methodology

Equations for path following and obstacle vector field guidance were presented. Singularities in a summed field were discussed and a method for locating them numerically was provided. A Dubin's UAV modeled representing a fixed wing UAV kinematics guided by a path following guidance is presented. Obstacles were defined in terms of UAV turning radius for convenience. Cost functions for evaluating the performance of obstacle avoidance in terms of path deviation was provided. Paths with strictly repulsive guidance were shown to guide UAV away from the obstacle but with excess path deviation, unnecessary turns, and slow path convergence. Adding circulation to the obstacle field improved performance and reduced the overall cost. A large space of circulation and decay multipliers were evaluated and the cost displayed in a heatmap. The combination of parameters that provide the least cost guidance can be selected to provide an optimized guidance, however takes significant computation time to evaluate the large parameter space. A numerical method for determining parameters was presented and an example of it's execution provided. Lastly, the optimized GVF guidance was implemented into python to be used for real-time obstacle avoidance guidance for the crazyflie quadcopter. Next, results for each phase will be discussed.

4 Results

4.1 Introduction to Results

fill out

4.2 Phase I & II

The methods discussed in Phase I and Phase II for detecting singularities in a GVF and optimizing obstacle field decay radius and circulation will now be demonstrated for several scenarios involving a fixed wing UAV. Avoidance scenarios that represent the possible configurations of an obstacle that lies along a pre-planned path consist of small obstacles, large obstacles, path centered obstacles, and path offset obstacles. small obstacles are considered those with radius $r_o = \theta_r$ while large obstacles have a radius $r_o \geq 5\theta_r$. Centered obstacles represent the worst case avoidance scenario where the UAV must deviate at least 50% of the obstacles radius in order to successfully avoid. For the path centered obstacle, counterclockwise and clockwise avoidance have identical costs. In the centered obstacle scenarios presented, it is assumed that clockwise avoidance is desired and the initial conditions and bounds are set accordingly. A table of the avoidance scenarios demonstrated is shown below in Table 4.1 below.

Table 4.1: GVF Avoidance Scenarios

Scenario Number	M	y_o
1	1	0
2	5	0
3	1	$0.5r_o$
4	5	$0.5r_o$

For each scenario a fixed wing UAV is assumed to be following a pre-planned path traveling at constant speed $u = 15m/s$ with a turnrate $\dot{\theta} = 20deg/s$. The UAV's initial position is set to $(100M, 0)$ and heading $\theta = 180°$ for each simulation to ensure a small length of the planned path is traveled prior to avoiding the obstacle.

Scenario 1 consists of a path centered obstacle with a radius equal to that of the UAV's turning radius θ_r. Minimizing the path deviation cost function $\bar{\gamma}$ in Phase II results in a route that avoids the obstacle and quickly returns to the planned path. The total cost of scenario 1 for avoiding the obstacle is $\bar{\gamma} = 7$ for decay radius multiplier $k = 2.8$ and circulation 2.1, shown in Figure 4.1.

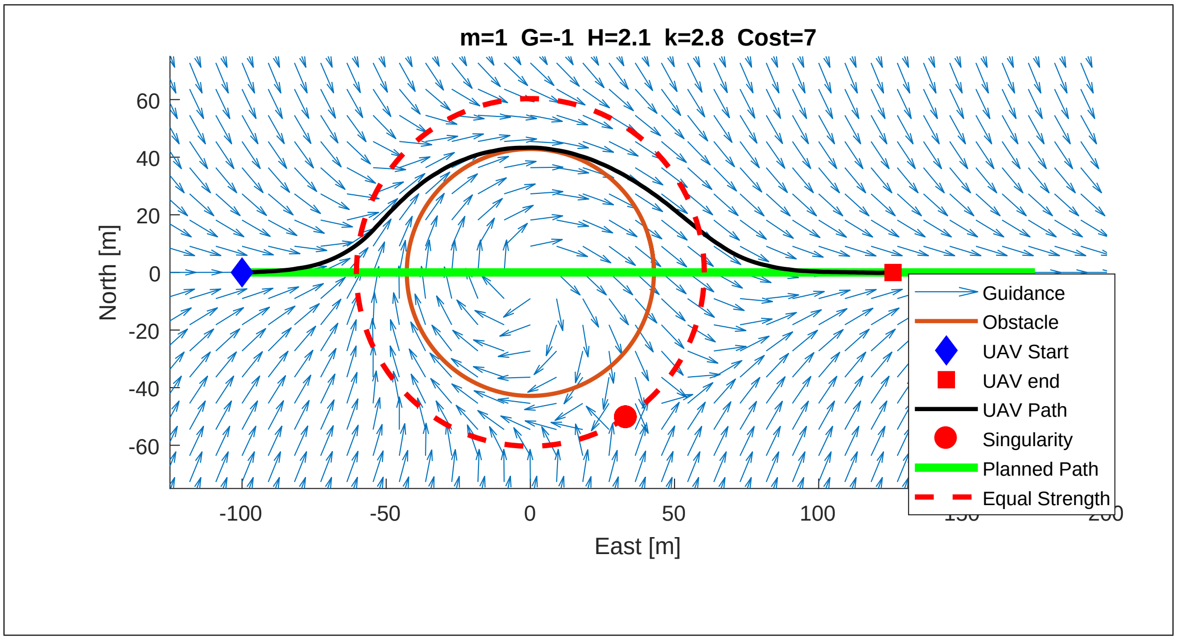


Figure 4.1:

Scenario 2 consists of a path centered obstacle with a radius equal to that of 5 times the UAV's turning radius θ_r. The total cost of scenario 2 for avoiding the obstacle is $\bar{\gamma} = 37$ for decay radius multiplier $k = 2.8$ and circulation 2.1, shown in Figure 4.1. Note that the optimized weights are identical to that of scenario I in this case.

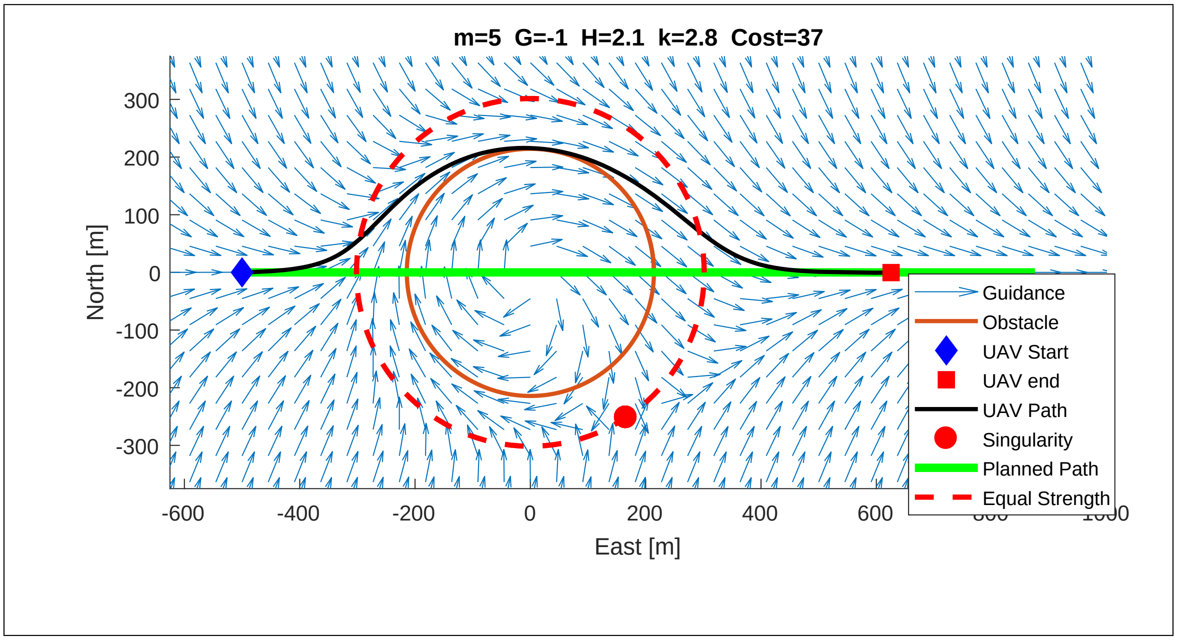


Figure 4.2:

Scenario 3 consists of a path off centered obstacle with a radius equal to that of the UAV's turning radius θ_r. The shortest direction around the obstacle can be determined from inspection, therefore negative initial conditions and bounds are given to the minimizer to produce the guidance shown in Figure 4.3. The total cost of scenario 3 for avoiding the obstacle is $\bar{\gamma} = 3$ for decay radius multiplier $k = 2.4$ and circulation $H_o = -2.6$.

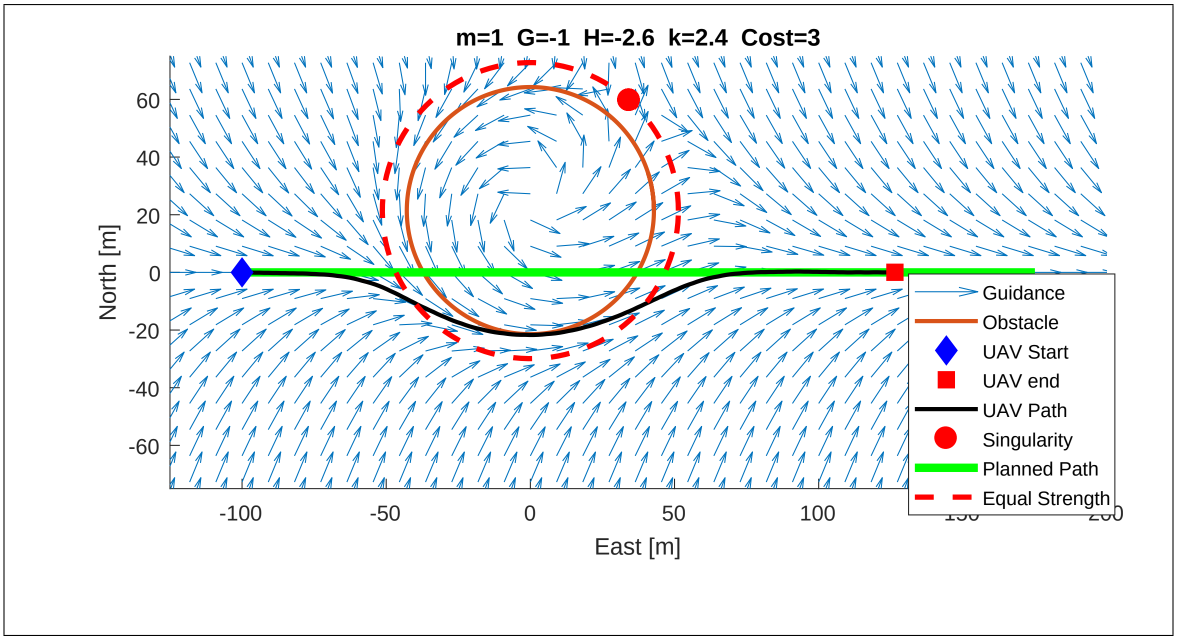


Figure 4.3:

Scenario 4 consists of a path off centered obstacle with a radius equal to 5 times that of the UAV's turning radius θ_r. Again, the shortest direction around the obstacle is determined from inspection. The total cost of scenario 4 for avoiding the obstacle is $\bar{\gamma} = 14$ for decay radius multiplier $k = 2.4$ and circulation $H_o = -2.7$.

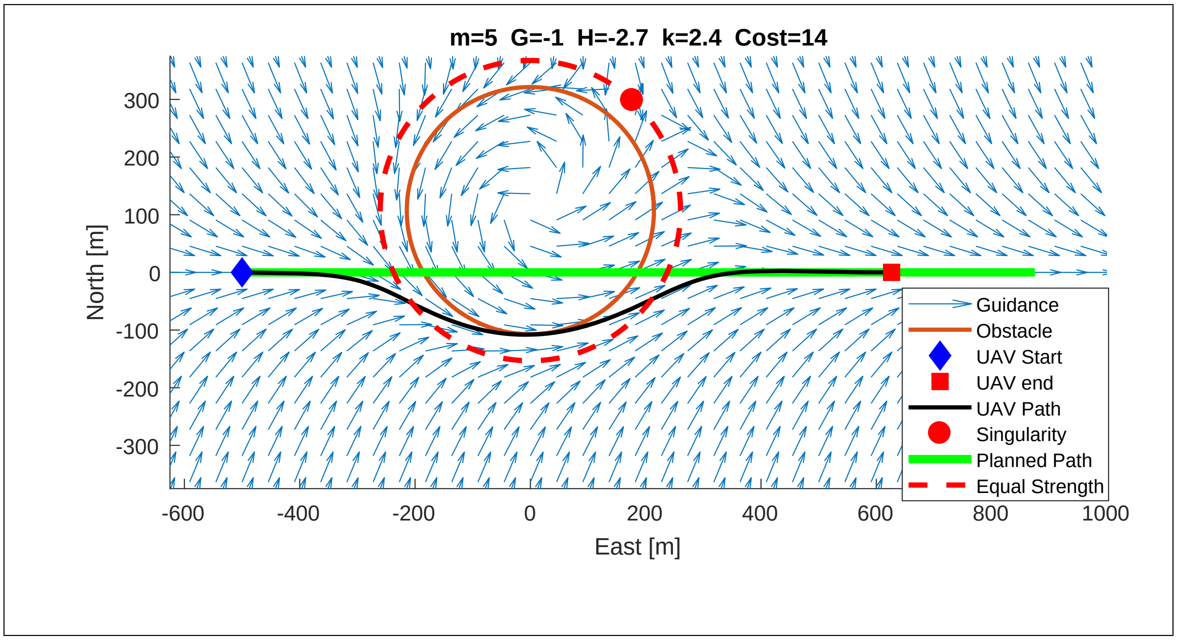


Figure 4.4:

4.3 Phase III

Optimized GVF guidance was programmed into a python written ground station to control a crazyflie 2.0 indoor quadcopter. Several obstacle scenarios presented in Phase II results were replicated and compared. Dubin's constraints were applied to the quadcopter to emulate fixed wing UAV dynamics in lieu of outdoor flight tests. Scenarios will be first presented followed by actual

5 Conclusions

UAVs typically rely on path planning algorithms to provide an obstacle free and flyable path prior to flight. In the event that unplanned obstacles are encountered a new path may have to be re-generated which may not be possible if communication with a ground station is lost. Path following and obstacle avoidance was achieved without the need to re-plan the mission path by using optimized GVF decay radius and circulation parameters. Simulations were conducted with a fixed wing UAV modeled as a Dubin's vehicle using the GVF guidance and circulation and decay radius were selected which minimized a path deviation cost function. Singularities in the GVF were characterized and located numerically. The optimized GVF guidance was implemented in a indoor quadcopter with turn-rate constraints to emulate a fixed wing UAV. Results comparing simulations and indoor flight tests were compared.

Guidance for path following and obstacle avoidance without the need to re-plan mission paths has the potential to aid in increasing unmanned systems autonomy. Obstacles can be avoided without the intervention of a human operator and they can do so at minimal deviation from the planned path. These planned paths represent a route in which a task must be completed, therefore, remaining close the path increases the UAVs overall effectiveness. The optimized GVF guidance promotes this increased mission effectiveness.

6 OLD

GVF guidance for following a straight path and avoiding circular obstacles was replicated from literature. A summed GVF for path following and circular obstacle avoidance was evaluated for GVF singularities. A method for determine the location of singularities in a summed GVF was presented. A worst case obstacle avoidance scenario with a circular obstacle centered on a straight path represented by a strictly repulsive field was evaluated for singularities. Circulation was added to the GVF to demonstrate mitigation of GVF singularities in a summed field.

6.0.1 Path Following Vector Field Guidance

Guidance for a path at angle $\delta = 0$ and equal parts circulation and convergence weights $G = H = 1$ is shown in Figure 6.1a. How quickly the path following field transitions from convergence to circulation depends on the field weights. Equal parts convergence and circulation are shown in Figure 6.1a ($G = H = 1$) and a larger circulation value in Figure 6.1b ($G = 1, H = 5$).

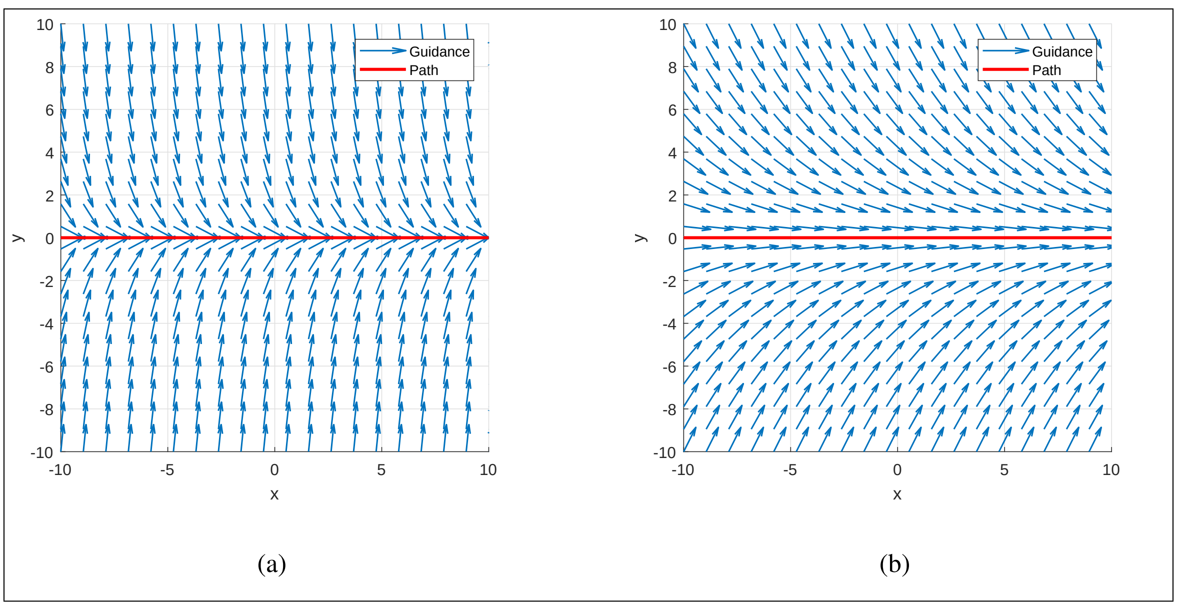
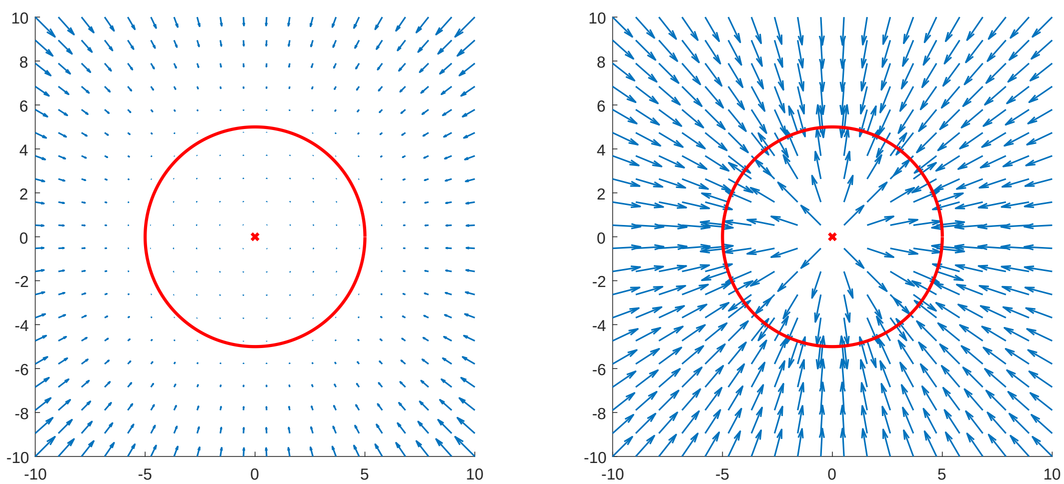


Figure 6.1: GVF converging and a) small circulation b) large circulation

6.0.2 Avoidance Vector Field Guidance

An obstacle field construction begins with the intersection of a cylinder and a plane. The non-normalized convergence guidance is shown in Figure 6.2a. Normalizing the convergence vectors is shown in Figure 6.2b. Note the the convergence vectors decay in magnitude as they approach the target curve. Normalizing the convergence vectors is done so that both convergence and circulation vectors are present at any range from the target path.



(a) (b)

Figure 6.2: GVF circular attractive field without normalization (a) and normalization (b)

Guidance that repels from a circular path can be produced by setting the convergence weight $G = -1$, shown in Figure 6.3. Inside of the target path, vectors point towards the center of the circle which may produce a trap situation if the UAV ends up inside the radius.

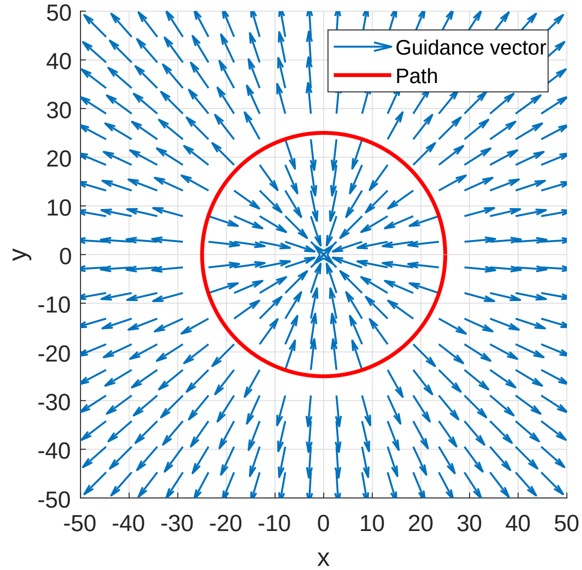


Figure 6.3: Repulsive Circular Field with Large Radius

To prevent trap situations, the radius of the target curve can be reduced several orders of magnitude compared to the actual obstacles radius. Reducing the path radius of Figure 6.3 results in a field that repels from what is effectively a small point, shown in Figure 6.4.

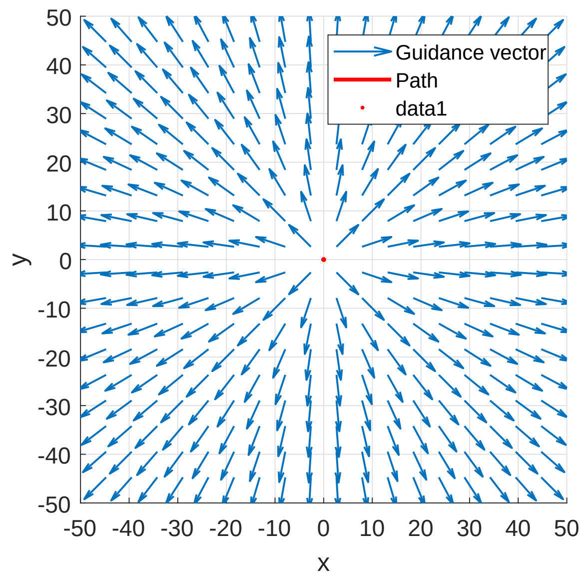


Figure 6.4: Repulsive Circular Field with Small Radius

Evaluating the circulation term in Equation 3.14 results in a vector field that is parallel to a circular path, shown in Figure 6.5a. The field is normalized to produce a field with equal length vectors for the configuration space which is shown in Figure 6.5b.

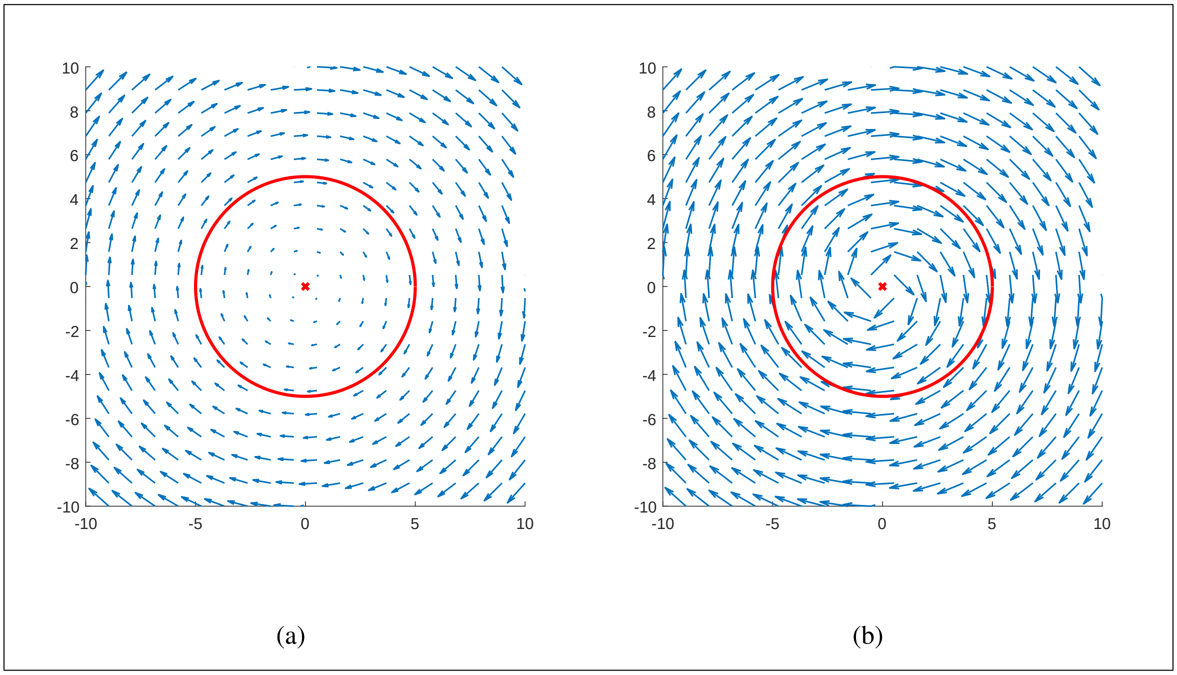
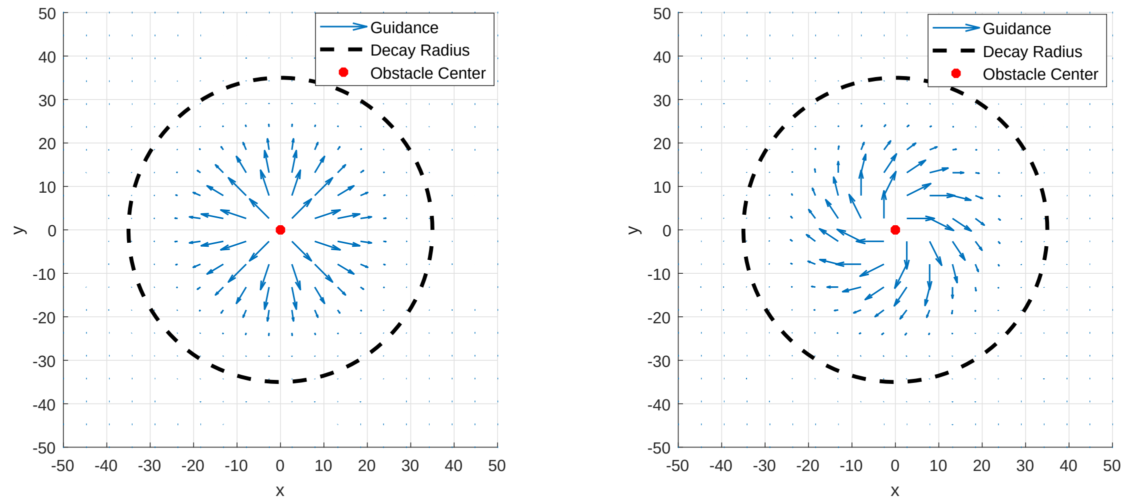


Figure 6.5: Circular GVF without normalization (a) and with normalization (b)

Repulsive GVFs should only act locally to repel UAVs away from obstacles to prevent deviation from the path prematurely. The guidance $\overrightarrow{V}_{||o||}$ multiplied by the decay function P is shown in Figure 6.6 with no circulation (a) and equal magnitude convergence and circulation (b).



(a) (b)

Figure 6.6: Repulsive GVF a) no circulation $H_o = 0$ and b) with circulation $H_o = 1$

7 FUTURE WORK

References

[1] Ariyur, K. B. and Fregene, K. O., "Autonomous tracking of a ground vehicle by a UAV," *American Control Conference, 2008*, IEEE, 2008, pp. 669–671.

[2] Teuliere, C., Eck, L., and Marchand, E., "Chasing a moving target from a flying UAV," *Intelligent Robots and Systems (IROS), 2011 IEEE/RSJ International Conference on*, IEEE, 2011, pp. 4929–4934.

[3] Frew, E. W., "Cooperative standoff tracking of uncertain moving targets using active robot networks," *Robotics and Automation, 2007 IEEE International Conference on*, IEEE, 2007, pp. 3277–3282.

[4] Griffiths, S., "Vector Field Approach for Curved Path Following for Miniature Aerial Vehicles," American Institute of Aeronautics and Astronautics, Aug. 2006.

[5] Goncalves, V. M., Pimenta, L. C. A., Maia, C. A., and Pereira, G. A. S., "Artificial vector fields for robot convergence and circulation of time-varying curves in n-dimensional spaces," IEEE, 2009, pp. 2012–2017.

[6] Gonçalves, V. M., Pimenta, L. C., Maia, C. A., Pereira, G. A., Dutra, B. C., Michael, N., Fink, J., and Kumar, V., "Circulation of curves using vector fields: actual robot experiments in 2D and 3D workspaces," *Robotics and Automation (ICRA), 2010 IEEE International Conference on*, IEEE, 2010, pp. 1136–1141.

[7] Gonçalves, V. M., Pimenta, L. C., Maia, C. A., Dutra, B. C., and Pereira, G. A., "Vector fields for robot navigation along time-varying curves in \$ n \$-dimensions," *IEEE Transactions on Robotics*, Vol. 26, No. 4, 2010, pp. 647–659.

[8] Panagou, D., "Motion planning and collision avoidance using navigation vector fields," *Robotics and Automation (ICRA), 2014 IEEE International Conference on*, IEEE, 2014, pp. 2513–2518.

[9] Bone, E., "UAVs backbround and issues for congress.pdf," 2003.

[10] Oh, H., Kim, S., Shin, H.-S., Tsourdos, A., and White, B., "Coordinated standoff tracking of groups of moving targets using multiple UAVs," *Control & Automation (MED), 2013 21st Mediterranean Conference on*, IEEE, 2013, pp. 969–977.

[11] Hyondong Oh, Seungkeun Kim, Hyo-sang Shin, and Tsourdos, A., "Coordinated standoff tracking of moving target groups using multiple UAVs," *IEEE Transactions on Aerospace and Electronic Systems*, Vol. 51, No. 2, April 2015, pp. 1501–1514.

[12] Wise, R. A. and Rysdyk, R. T., "UAV coordination for autonomous target tracking," *Proceedings of the AIAA Guidance, Navigation, and Control Conference, Keystone, CO, Aug*, 2006, pp. 21–24.

[13] Ulun, S. and Unel, M., "Coordinated motion of UGVs and a UAV," *Industrial Electronics Society, IECON 2013-39th Annual Conference of the IEEE*, IEEE, 2013, pp. 4079–4084.

[14] Beard, R. W. and McLain, T. W., *Small unmanned aircraft: theory and practice*, Princeton University Press, Princeton, N.J, 2012, OCLC: ocn724663112.

[15] Technology, C. C., "," 2018.

[16] Martin, L., "Kestral Flight Systems And Autopilot," 2018.

[17] DroneCode, "Pixhawk 1 Flight Controller Guide," 2018.

[18] Nelson, D. R., "Cooperative control of miniature air vehicles," 2005.

[19] Nelson, D. R., Barber, D. B., McLain, T. W., and Beard, R. W., "Vector field path following for small unmanned air vehicles," *American Control Conference, 2006*, IEEE, 2006, pp. 7–pp.

[20] Nelson, D., Barber, D., McLain, T., and Beard, R., "Vector Field Path Following for Miniature Air Vehicles," *IEEE Transactions on Robotics*, Vol. 23, No. 3, June 2007, pp. 519–529.

[21] Manjunath, A., Mehrok, P., Sharma, R., and Ratnoo, A., "Application of virtual target based guidance laws to path following of a quadrotor UAV," IEEE, June 2016, pp. 252–260.

[22] Sujit, P., Saripalli, S., and Sousa, J. B., "Unmanned Aerial Vehicle Path Following: A Survey and Analysis of Algorithms for Fixed-Wing Unmanned Aerial Vehicless," *IEEE Control Systems*, Vol. 34, No. 1, Feb. 2014, pp. 42–59.

[23] Khatib, O., "Real-time obstacle avoidance for manipulators and mobile robots," *The international journal of robotics research*, Vol. 5, No. 1, 1986, pp. 90–98.

[24] Rimon, E., "Exact Robot Navigation Using Artificial Potential Functions.pdf," 1992.

[25] Liu, Y. and Zhao, Y., "A virtual-waypoint based artificial potential field method for UAV path planning," *Guidance, Navigation and Control Conference (CGNCC), 2016 IEEE Chinese*, IEEE, 2016, pp. 949–953.

[26] Borenstein, J. and Koren, Y., "Real-time obstacle avoidance for fast mobile robots in cluttered environments," *Robotics and Automation, 1990. Proceedings., 1990 IEEE International Conference on*, IEEE, 1990, pp. 572–577.

[27] Borenstein, J. and Koren, Y., "The vector field histogram-fast obstacle avoidance for mobile robots," *IEEE transactions on robotics and automation*, Vol. 7, No. 3, 1991, pp. 278–288.

[28] Koren, Y. and Borenstein, J., "Potential Field Methods and their inherent limitions for morbile robot navigation.pdf," 1991.

[29] Kim, D. H., "Escaping route method for a trap situation in local path planning," *International Journal of Control, Automation and Systems*, Vol. 7, No. 3, June 2009, pp. 495–500.

[30] Goerzen, C., Kong, Z., and Mettler, B., "A Survey of Motion Planning Algorithms from the Perspective of Autonomous UAV Guidance," *Journal of Intelligent and Robotic Systems*, Vol. 57, No. 1-4, Jan. 2010, pp. 65–100.

[31] Lei Tang, Songyi Dian, Gangxu Gu, Kunli Zhou, Suihe Wang, and Xinghuan Feng, "A novel potential field method for obstacle avoidance and path planning of mobile robot," IEEE, July 2010, pp. 633–637.

[32] Li, G., Yamashita, A., Asama, H., and Tamura, Y., "An efficient improved artificial potential field based regression search method for robot path planning," IEEE, Aug. 2012, pp. 1227–1232.

[33] Jung, W., Lim, S., Lee, D., and Bang, H., "Unmanned Aircraft Vector Field Path Following with Arrival Angle Control," *Journal of Intelligent & Robotic Systems*, Vol. 84, No. 1-4, Dec. 2016, pp. 311–325.

[34] Frew, E., "Lyapunov Guidance Vector Fields for Unmanned Aircraft Applications.pdf," .

[35] Chen, H., Chang, K., and Agate, C. S., "Tracking with UAV using tangent-plus-Lyapunov vector field guidance," *Information Fusion, 2009. FUSION'09. 12th International Conference on*, IEEE, 2009, pp. 363–372.

[36] Chen, H., Chang, K., and Agate, C. S., "UAV path planning with tangent-plus-Lyapunov vector field guidance and obstacle avoidance," *IEEE Transactions on Aerospace and Electronic Systems*, Vol. 49, No. 2, 2013, pp. 840–856.

[37] Liang, Y. and Jia, Y., "Tangent vector field approach for curved path following with input saturation," *Systems & Control Letters*, Vol. 104, June 2017, pp. 49–58.

[38] Pereira, G. A. S., Choudhury, S., and Scherer, S., "A framework for optimal repairing of vector field-based motion plans," IEEE, June 2016, pp. 261–266.

[39] Pimenta, L. C., Pereira, G. A., and Mesquita, R. C., "Fully continuous vector fields for mobile robot navigation on sequences of discrete triangular regions," *Robotics and Automation, 2007 IEEE International Conference on*, IEEE, 2007, pp. 1992–1997.

[40] Md, Z., Rg, C., and Dj, G., "Simplex Solutions for Optimal Control Flight Paths in Urban Environments," *Journal of Aeronautics & Aerospace Engineering*, Vol. 06, No. 03, 2017.

[41] Zhou, D. and Schwager, M., "Vector field following for quadrotors using differential flatness," IEEE, May 2014, pp. 6567–6572.

[42] Gerlach, A. R., *Autonomous Path-Following by Approximate Inverse Dynamics and Vector Field Prediction*, University of Cincinnati, 2014.