Faculty of Aerospace Engineering

Relative Localization for Collision Avoidance in Micro Air Vehicle teams

Using on-board processing and sensors in indoor environments

M. Coppola

June 20, 2016

Delft University of Technology

Challenge the future

Relative Localization for Collision Avoidance in Micro Air Vehicle teams

Using on-board processing and sensors in indoor environments

Master of Science Thesis

For obtaining the degree of Master of Science in Aerospace Engineering
at Delft University of Technology

M. Coppola

June 20, 2016

Faculty of Aerospace Engineering · Delft University of Technology

Delft University of Technology

Delft University Of Technology
Department Of
Control and Simulation

The undersigned hereby certify that they have read and recommend to the Faculty of Aerospace Engineering for acceptance a thesis entitled **"Relative Localization for Collision Avoidance in Micro Air Vehicle teams"** by **M. Coppola** in partial fulfillment of the requirements for the degree of **Master of Science**.

Dated: June 20, 2016

Readers:

Dr. Q.P. Chu

Dr. G.C.H.E. de Croon

Ir. K.N. McGuire

Dr. M. Mazo Jr.

Abstract

Teams of quadrotors can be used for surveillance, mapping, and measurement tasks. A current limitation is the high risk of collisions between members when flying within a confined space. The MAVs need knowledge of their relative location in order to perform evasive maneuvers. In new, unknown, and GPS-denied environments, this is something that the MAVs must measure on-board.

Bluetooth is a low mass and low power technology readily available on even the smallest MAVs. It can be used for inter-MAV communication during team tasks. When communicating, Bluetooth antennas measure the power of the received signal. This is correlated with the distance between the transmitter and the receiver. This thesis builds on this concepts and proposes an on-board, Bluetooth enabled, relative localization scheme. On-board states (velocity, height, and planar orientation) are communicated directly between drones using a Bluetooth connection. This is fused with the range inferred from the signal strength to obtain a 3D relative location estimate. A relative collision avoidance controller is then proposed that is specifically designed to deal with the expected performance of the localization scheme. The relative collision avoidance strategy is based on collision cones, of which the size is tailored to encompass the expected localization errors. Evasive maneuvers are selected using a clock-wise search in order to provide a reciprocal scenario.

The system was tested with a team of AR-Drones 2.0 flying in a 4m×4m arena. The enforced task requested the AR-Drones to repeatedly fly from wall to wall whilst passing through the center of the arena, hence making collisions highly likely. When using two AR-Drones and off-board velocity/orientation estimates, the drones are able to fly around the arena and avoid each other for the entire flight time as permitted by the battery. With three ARDrones under the same conditions, the flight time to collision was approximately 3 minutes. With two ARDrones flying with on-board velocity estimation, the time to collision was approximately 3 minutes due to the additional disturbances in velocity estimates. A simulation environment has been set-up to test the merits of the collision avoidance scheme in different configurations regarding MAV diameter and arena size. It is shown that improvements in RSSI sensor noise and the use of smaller MAVs can significantly improve the outcome.

Aknowledgements

I would like to thank: Dr. Guido de Croon, for entrusting me with the project and leading me throughout it; Kimberly McGuire, for the invaluable support and guidance; Kirk Scheper, for his help as well as the provision of Bluetooth communication between the drones.

M. Coppola
Delft, June 2016

Acronyms

3D	3-Dimensional
AOA	Angle of Arrival
AP	Access Point
CALU	Collision Avoidance under Localization Uncertainty
CC	Collision Cone
CG	Center of Gravity
DOF	Degrees of Freedom
EKF	Extended Kalman Filter
f.o.r.	frame of reference
FSL	Free Space Loss
GPS	Global Positioning System
GT	Ground-Truth
HL	Human-Like
HRVO	Hybrid Reciprocal Velocity Obstacle
IAEKF	Iterative Adaptive EKF
IMU	Inertial Measurement Unit
IR	Infra-Red
ISM	Industrial, Scientific and Medical
KCF	Kalman Consensus Filter
KF	Kalman Filter
LD	Log-Distance
LED	Light-Emitting Diode
LoG	Laplacian of Gaussian
LQI	Link Quality Indicator
LS	Least Squares
MAF	Moving Average Filter
MAV	Micro Areal Vehicle

MSL	Mean Sea Level
NED	North-East-Down
OF	Optical Flow
ORCA	Optimal Reciprocal Collision Avoidance
PF	Particle Filter
RANSAC	RANdom SAmpling and Consensus
RLS	Recursive Least Squares
RMSE	Root Mean Squared Error
ROS	Robotics Operating System
RSS	Received Signal Strength
RSSI	Received Signal Strength Indication
RTOA	Round-trip Time of Arrival
RVO	Reciprocal Velocity Obstacle
SIDPAC	System Identification Programs for Aircraft
SIFT	Scale-Invariant Feature Transform
SLAM	Simultaneous Localization and Mapping
STDMA	Self-Organized Time Division Multiple Access
SURF	Speeded Up Robust Feature
TDOA	Time Difference of Arrival
TOA	Time of Arrival
UAV	Unmanned Aerial Vehicle
VO	Velocity Obstacle
WLAN	Wireless Local Area Network
WSN	Wireless Sensor Network
ZMGN	Zero-Mean Gaussian Noise

Contents

Part I

Scientific Paper

On-board Bluetooth-based Relative Localization for Collision Avoidance in Micro Air Vehicle teams

M. Coppola, K.N. McGuire, K.Y.W. Scheper, and G.C.H.E. de Croon[1]

Abstract— **A current limitation of using Micro Air Vehicles in teams is the high risk of collisions between members. Knowledge of relative location is needed in order to perform evasive maneuvers from such collisions. We propose an on-board Bluetooth-based relative localization scheme. Bluetooth is a light-weight and energy efficient communication technology that is readily available on even the smallest Micro Air Vehicle units. In this work, it is exploited for communication between team members to exchange on-board states (velocity, height, and orientation), and the strength of the communication signal is used to infer relative range. The data is fused on-board by each Micro Air Vehicle to obtain a relative estimate of the location and motion of all other team members. Furthermore, a collision avoidance controller is proposed based on collision cones. It is designed to deal with the expected performance of the localization scheme by adapting the collision cones during flight and enforcing a clock-wise evasion maneuver. The system was tested with a team of AR-Drones 2.0 flying in a 4m×4m arena. The task requested the AR-Drones to repeatedly fly from wall to wall whilst passing through the center of the arena, hence making collisions highly likely. The system showed promising results. When using two AR-Drones and off-board velocity/orientation estimates, the drones are able to fly around the arena and avoid each other for the entire flight time as permitted by the battery. With three AR-Drones under the same conditions, flight time to collision was 3 minutes. With two AR-Drones flying with on-board velocity estimation, the time to collision was approximately 3 minutes due to the disturbances in velocity estimates. Simulation results show that significantly better results can be expected with smaller units.**

I. INTRODUCTION

Micro Air Vehicles (MAVs) applications include: surveillance and mapping [1] [2], and visual and/or chemical inspection of forest fires and disaster areas [3] [4] [5] [6]. To push the boundaries of such applications, state-of-the-art technology has led to miniaturized variants such as the Lisa-S Ladybird [7] or the Pico-Quadrotor [8]. These platforms benefit from: lower mass, increased portability, less obtrusive/restricted navigation (valuable for indoor environments), and safer use near humans. Allowing several of these MAVs to operate in a homogeneous team improves performance by reducing the task execution time and adding redundancy, scalability, and versatility [9] [2].

When a team of homogeneous MAVs with decentralized control performs an arbitrary task in a confined indoor space (e.g. a room), there is a non-negligible risk of inter-member collisions [10]. This is a failure condition to be avoided to ensure mission success. Albeit a team behavior can emerge without inter-member awareness [11], collision avoidance requires on-board knowledge by each MAV of the relative location of the other team members. Additionally, knowledge of relative location can empower more complex team behaviors such as leader-follower [8] and formation-flying [12].

A method to provide all MAVs with relative localization estimate is to rely on a global (shared) reference frame in which each MAV can localize itself. The MAVs can then communicate, compare their position co-ordinates, and infer a relative estimate. In outdoor tasks, Global Positioning System (GPS) receivers can be used to obtain global position data that is then shared [13][14], but these do not function indoors [15]. In order to achieve the same effect without compromising the aforementioned efforts of MAV miniaturization, several solutions propose planting external sensors/beacons with known (relative) locations, such as: motion tracking cameras [16], fixed wireless transmitters/receivers [17], or fixed visual markers [18]. Although effective, these solutions defeat the purpose of several exploratory tasks by relying on a pre-arranged environment. The Simultaneous Localization and Mapping (SLAM) strategy and its variants attempt to solve this by defining a map on-board during flight [2]. However, when map generation is not part of the specific mission, then this is a resource intensive practice to be discouraged [19], or even beyond the current capabilities of miniaturized MAVs [7].

The use of direct MAV-to-MAV measurements overcomes these disadvantages. Vision has received attention due to its generally favorable end-results, but examples found in literature simplify the detection task with the adoption of mounted visual aids in the form of: (red) balls [20], tags [21], or markers [22]. These studies benefit from the combination of relatively high-resolution cameras, fast processing speeds, and large markers if compared to more miniaturized drones. However, test results in the exploratory phases of this study have led to the conclusion that using vision without such aids and at lower resolutions (128×96px, as seen on the Lisa-S Ladybird [23] [24]) becomes highly problematic and prone to false-positives/false-negatives. Furthermore, performance is dependent on lighting conditions, which may change. Other disadvantages of using vision are: the need for a front-facing camera and flight along its axis, limited

[1]Micro Air Vehicle Laboratory, Department of Control and Simulation, Faculty of Aerospace Engineering, TU Delft Robotics Institute, Delft University of Technology, Kluyverweg 1, 2629 HS Delft, The Netherlands `g.c.h.e.decroon@tudelft.nl`

field-of-view, and generally high processing requirements. Infra-Red (IR) sensors have been proposed as an alternative but these require multiple units to be arranged in a rigid structure, leading to a high mass and power consumption penalty [25].

The work in [26] attempts to overcome the issues above by using on-board sound-based localization with a mounted microphone array. The difference between arrival times at the different receivers is used to infer the relative bearing. A passive version of this attempted to locate propeller sounds of nearby MAVs, yet this suffered from noise coming from the MAV's own engine. The issue was overcome by the introduction of a chirp generator to send audio signals at specific frequencies [27]. Nevertheless, the method still suffers from noise induced by wind (which is unavoidable on moving MAVs) and structural vibrations which compromise the rigidity of the microphone array [28]. Furthermore, a dedicated microphone array and chirp generator is needed. For smaller MAVs, this can account for an increase in mass of 10%-20% [28] [7] and it may limit the scalability of the approach.

Bluetooth is a technology that is readily available at a low mass, power, and cost penalty even on the smaller MAV units [29] [23]. It may be used for inter-member communication [10]. Bluetooth communication natively measures a quantity known as Received Signal Strength Indication (RSSI), which is a measure (in dB) of the power in the received signal [30] [31]. This decreases with distance and can be used as a measure for inter-MAV range. This knowledge has already been exploited by attempting collision avoidance using only relative range sensing [10]. However, the significant noise and disturbances experienced, coupled with the use of range-only measurements, were insufficient to guarantee safe flight of two or more MAVs in a confined area.

This article introduces an *on-board relative localization* method for MAVs based on Bluetooth communication, and proposes a tailored relative avoidance strategy to be used. The method relies on Bluetooth as a measure of relative distance (via RSSI) *and* as a method for the exchange of own state measurements in order for each MAV to estimate the relative location of nearby team-members. The advantages of this solution are: a) it provides a continuous localization estimate at all relative bearings, such that the drone is not forced to face any particular direction; b) it has a low dependence on the lighting and sound conditions of the environment; c) it has low energy consumption requirements; d) it does not require the addition of any dedicated sensors, so mass is unaffected. A reactive collision avoidance strategy is proposed that relies on the localization estimates and is designed to deal with the expected errors. It is based on the concept of collision cones [32] [33].

The remainder of this article is organized as follows. Section II introduces Bluetooth RSSI and how it is combined with own/received state measurements in order to obtain a relative localization estimate. Section III describes the collision avoidance strategy implemented in the MAVs. Tests have been performed in simulation and in the real-world using Bluetooth equipped AR-Drones 2.0 to determine the performance of the system with respect to collision avoidance. The test environments and methodologies used are defined in Section IV. The results are described in Section V and further discussed in Section VI. Section VII provides concluding remarks.

II. BLUETOOTH-BASED RELATIVE LOCALIZATION

A. Framework Definition for Relative Localization

Consider two MAVs $\mathcal{R}_i$ and $\mathcal{R}_j$ with right-handed body-fixed frames $\mathcal{F}_{B_i}$ and $\mathcal{F}_{B_j}$, respectively. Under this framework, the relative location of $\mathcal{R}_j$ with respect to $\mathcal{R}_i$ can be defined as:

$$\vec{\mathcal{P}}_{ji} = \begin{bmatrix} x_{ji} & y_{ji} & z_{ji} & \phi_{ji} & \theta_{ji} & \psi_{ji} \end{bmatrix}. \tag{1}$$

$\vec{\mathcal{P}}_{ji}$ is a vector of the quantities describing the 3-Dimensional (3D) relative pose of $\mathcal{R}_j$ with respect to $\mathcal{R}_i$. It is expressed in $\mathcal{F}_{B_i}$. x_{ji}, y_{ji}, and z_{ji} are the position of the origin of $\mathcal{R}_j$ in $\mathcal{F}_{B_i}$; ϕ_{ji}, θ_{ji}, and ψ_{ji} are the roll, pitch, and yaw angles of $\mathcal{F}_{B_j}$ with respect to $\mathcal{F}_{B_i}$.

Pitch and roll may be neglected by assuming that quadrotors maintain approximately planar orientations with respect to the ground. Equation (1) may then be re-defined with polar coordinates as:

$$\vec{\mathcal{P}}_{ji} = \begin{bmatrix} \rho_{ji} & \beta_{ji} & h_{ji} & \psi_{ji} \end{bmatrix}. \tag{2}$$

ρ_{ji} represents the range between the origins of $\mathcal{F}_{B_i}$ and $\mathcal{F}_{B_j}$, see Equation (3). β_{ji} is the horizontal planar bearing of the origin of $\mathcal{F}_{B_j}$ with respect to $\mathcal{F}_{B_i}$, see Equation (4). h_{ji} is the height of $\mathcal{R}_j$ with respect to $\mathcal{R}_i$, see Equation (5). ψ_{ji} is the orientation of $\mathcal{R}_j$ with respect to $\mathcal{R}_i$, see Equation (6). The framework is depicted in Figure 1.

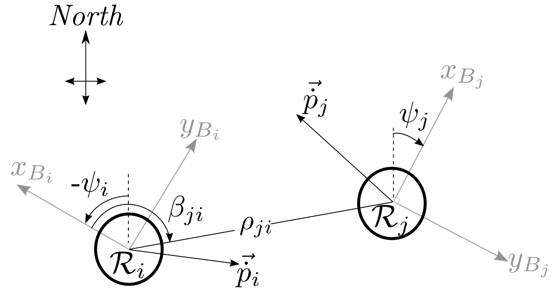


Fig. 1: Top view of relative localization framework of MAV $\mathcal{R}_j$ by MAV $\mathcal{R}_i$. x_B and y_B are the axis of the body frame $\mathcal{F}_B$, where x_B is positive in the forwards direction. z_B is positive down-wards (into the page). The orientation of the body frames is with respect to North.

$$\rho_{ji} = \sqrt{x_{ji}^2 + y_{ji}^2 + h_{ji}^2} \tag{3}$$

$$\beta_{ji} = atan2(y_{ji}, x_{ji}) \tag{4}$$

B. Relative Localization via Fusion of Range and Shared States

Achieving a relative localization estimate requires measuring or inferring the quantities in $\vec{\mathcal{P}}_{ji}$. Thanks to inter-drone communication, h_{ji} and ψ_{ji} are trivially observable by an MAV $\mathcal{R}_i$ by taking the difference between its own on-board state and the received states from $\mathcal{R}_j$:

$$h_{ji} = h_j - h_i, \tag{5}$$

$$\psi_{ji} = \psi_j - \psi_i. \tag{6}$$

ψ_i and ψ_j are the rotations of $\mathcal{F}_{B_i}$ and $\mathcal{F}_{B_j}$ with respect to a common reference axis. The common axis, as depicted in Figure 1, is magnetic North. It is discernible by all MAVs using a magnetometer [34] [35]. h_j and h_i are the height of the origin of $\mathcal{F}_{B_i}$ and $\mathcal{F}_{B_j}$ with respect to a reference height (e.g. Mean Sea Level (MSL), ground). This may be measured on-board using, for instance, a pressure-sensor [36] [37] [38], and/or a downward facing camera [39] [40] if assuming that the ground is flat.

ρ_{ji} is measured with RSSI measurements, which are correlated with range as elaborated in Section II-C. When h_{ji}, ψ_{ji}, and ρ_{ji} are measured, the relative bearing β_{ji} is observable [41] [42].

We use a discrete-time Extended Kalman Filter (EKF) in order to perform sensor fusion and observe β_{ji}. With potential implementation on more minimalistic systems in mind, the EKF is chosen due to its efficient processing and memory requirements [43]. Let Equation (7) be the process update equation between time step k and $k+1$ of the EKF.

$$\begin{bmatrix} \vec{p}_{ji} \\ \dot{\vec{p}}_i \\ \dot{\vec{p}}_{jRi} \\ \psi_j \\ \psi_i \\ h_j \\ h_i \end{bmatrix}_{k+1} = \begin{bmatrix} \vec{p}_{ji} + \left(\dot{\vec{p}}_{jRi} - \dot{\vec{p}}_i\right)\Delta t \\ \dot{\vec{p}}_i \\ \dot{\vec{p}}_{jRi} \\ \psi_j \\ \psi_i \\ h_j \\ h_i \end{bmatrix}_k + \mathbf{Q} \tag{7}$$

$\vec{p}_{ji} = \begin{bmatrix} x_{ji} & y_{ji} \end{bmatrix}^T$ holds Cartesian estimates of bearing and range. $\dot{\vec{p}}_i = \begin{bmatrix} \dot{x}_i & \dot{y}_i \end{bmatrix}^T$ is a vector of the velocity of $\mathcal{R}_i$ in $\mathcal{F}_{B_i}$ (see Figure 1). $\dot{\vec{p}}_{jRi}$ is $\dot{\vec{p}}_j$ *rotated* from $\mathcal{F}_{B_j}$ to $\mathcal{F}_{B_i}$. Δt is a discrete time step between updates equal to the time between k and $k+1$. $\mathbf{Q}$ is the process noise matrix, discussed further in Section II-D. This update equation assumes that all current velocities and headings remain constant. The measurement equation of the EKF is given by Equation (8).

$$\begin{bmatrix} m_{\rho_{ji}} \\ \dot{\vec{p}}_i \\ \dot{\vec{p}}_j \\ \psi_j \\ \psi_i \\ h_j \\ h_i \end{bmatrix}_{k+1} = \begin{bmatrix} \mathcal{L}(\rho_{ji}) \\ \dot{\vec{p}}_i \\ \mathbf{R_{2D}}(\psi_{\mathbf{ji}}) \cdot \dot{\vec{p}}_{jRi} \\ \psi_j \\ \psi_i \\ h_j \\ h_i \end{bmatrix}_k + \mathbf{R} \tag{8}$$

$m_{\rho_{ji}}$ is a measurable quantity that is correlated with ρ_{ji}. In this case $m_{\rho_{ji}}$ is the measured RSSI in dB during communication, which is a function of the range ρ_{ji} as given by $\mathcal{L}(\rho_{ji})$, see Equation (9). $\mathbf{R_{2D}}(\cdot)$ is a 2D rotation matrix. It makes use of the relative heading ψ_{ji} to rotate $\dot{\vec{p}}_j$ from $\mathcal{F}_{B_j}$ to $\mathcal{F}_{B_i}$. $\mathbf{R}$ is the relevant measurement noise matrix, discussed further in Section II-D.

It is noted that this scheme suffers from a degenerate motion known as *rotation ambiguity* [44]. When the path of $\mathcal{R}_j$ perfectly matches the path by $\mathcal{R}_i$ in a straight line, range-only measurements remain constant and are not informative for bearing estimation. For randomly flying MAVs, the probability of this event is negligible [44]. The same effect takes place when both $\mathcal{R}_i$ and $\mathcal{R}_j$ are static. Motion by at least one entity is required.

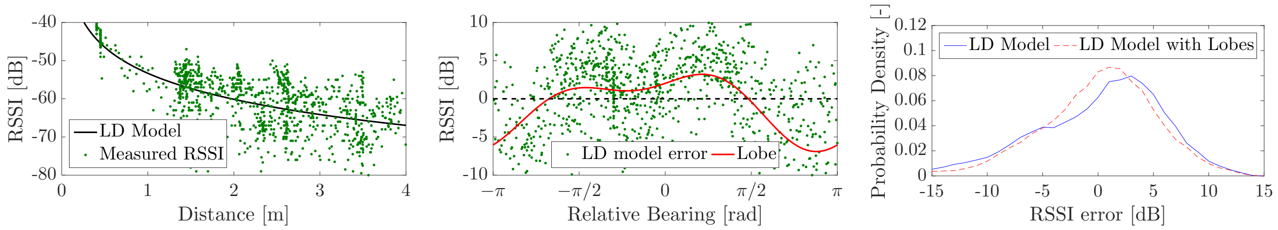
C. Signal Strength as a Range Measurement

In the EKF, the range measurement $m_{\rho_{ji}}$ is RSSI, which is a function $\mathcal{L}(\rho_{ji})$ of the signal power loss over a distance ρ_{ji}. Power loss of a signal may be modeled according to the Log-Distance (LD) model [45]:

$$\mathcal{L}(\rho_{ji}) = P_n - 10 \cdot \gamma_l \cdot \log_{10}(\rho_{ji}). \tag{9}$$

In Equation (9): P_n is the total power loss in dB at a nominal distance of $1m$, and γ_l is the *space-loss parameter*. It dictates the decay of the signal's power with distance. For free-space: $\gamma_l = 2.0$. Experimentally, it has been found that office buildings can feature $2 \leq \gamma_l \leq 6$ [46]. A sensitivity analysis of the model showed that an accurate identification of γ_l has a low impact on the distance estimate at lower distances (which is the scope of this article). The LD model is generally assumed subject to a Zero-Mean Gaussian Noise (ZMGN) [47] [48].

We analyzed the LD with a Bluetooth-enabled Ladybird MAV [7] and a fixed omni-directional Bluetooth antenna (W1049B by Pulse [49]). The Ladybird MAV was carried in concentric circles at different distances around the antenna. Its heading was kept constant so as to vary relative bearing throughout the measurements. The results from a representative data-sample are presented in Figure 2, to which the LD model is fitted using a non-linear Least Squares (LS) estimator as in Figure 2a. The standard deviation of the error about the model was found to be between $3dB$ and $6dB$. This is in line with the findings from [10] and [50].



(a) RSSI measurements with respect to distance (green-dotted) and fitted Log-Distance model (black, solid).

(b) Error of LD model with respect to relative bearing (green, dotted) fitted with a second order Fourier series (red, solid).

(c) Noise about the LD model without (blue, solid) and with (red, dashed) lobe impact.

Fig. 2: Results of RSSI measurements during an experiment whereby a Bluetooth-enabled Ladybird MAV was carried in circles around a fixed Bluetooth antenna. The heading of the Ladybird MAV was kept constant so as to explore the impact of relative bearing on the model error.

Figure 2b shows the error of the LD model as a function of the relative bearing. The presence of antenna lobes is discerned. This knowledge suggests an extension of the LD model with an additional gain term that is a function of relative bearing [10], but this is rejected. Including a term that is dependent on bearing means that a change in RSSI can be ambiguously associated to a change in either bearing or range. This negatively affects the convergence of the EKF. Furthermore, the shape of the lobes was found to be antenna-specific when tested with different MAVs, requiring an inconvenient increase in calibration efforts if lobes were to be included. Figure 2c shows that the error distribution is also not perfectly Gaussian. The main reason for this is deemed to be the presence of antenna lobes. A model that includes them can be seen to feature a more Gaussian distribution, although a positive skew is still present. Other disturbances may be: the interference by the reflection of the signal in the environment [45] [48], the presence of other signals in the $2.4GHz$ spectrum [46] [51], other objects that obstruct the signal [46].

D. Tuning of Noise Covariance Matrices

This section discusses the chosen tuning of the EKF covariance matrices $\mathbf{R}$ and $\mathbf{Q}$. The measurement noise matrix $\mathbf{R}$ is a diagonal matrix with the form shown in Equation (10).

$$\mathbf{R} = \begin{bmatrix} \sigma_m^2 & & & \\ & \sigma_v^2 \cdot \mathbf{I_{4\times4}} & & \\ & & \sigma_\psi^2 \cdot \mathbf{I_{2\times2}} & \\ & & & \sigma_h^2 \cdot \mathbf{I_{2\times2}} \end{bmatrix}. \tag{10}$$

σ_m is the expected standard deviation of the noise on $m_{\rho_{ji}}$. σ_v is the expected standard deviation of the noise on $\dot{\vec{p}}_i$ and $\dot{\vec{p}}_j$. σ_ψ is the expected standard deviation of the magnetic orientation measurements. σ_h is the expected standard deviation of the height measurements. $\mathbf{I_{n\times n}}$ is an $n \times n$ identity matrix such that the same standard deviation transfers to the relevant variables.

Considering the noise analysis of the LD model discussed in Section II-C, a foreseen disadvantage of using the EKF is the assumption of Gaussian noise on the RSSI measure. The effects are limited by adopting a high standard deviation for the received RSSI, therefore σ_m is tuned to $5dB$. All other variables in $\mathbf{R}$ can be tuned according to the expected noise from the on-board estimates.

The process noise matrix $\mathbf{Q}$ is the diagonal matrix presented in Equation (11). It needs to be tuned so as to define the validity of the expected process [52].

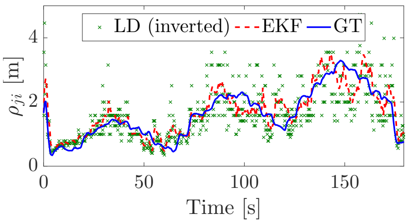
$$\mathbf{Q} = \begin{bmatrix} \sigma_{Q_p}^2 \cdot \mathbf{I_{2\times2}} & & & \\ & \sigma_{Q_v}^2 \cdot \mathbf{I_{4\times4}} & & \\ & & \sigma_{Q_\psi}^2 \cdot \mathbf{I_{2\times2}} & \\ & & & \sigma_{Q_h}^2 \cdot \mathbf{I_{2\times2}} \end{bmatrix}. \tag{11}$$

σ_{Q_p} is the standard deviation of the process noise on the relative position update. σ_{Q_v}, σ_{Q_ψ}, and σ_{Q_h} are the process noises for the expected updates in velocity, orientation, and height respectively. The tuning is made such that a high-level of trust is put on the relative position update, whereas lower trust is put on the update of the other quantities. This promotes convergence towards a bearing estimate and helps to discard the high noise and disturbance in the RSSI measurements. The values are tuned to the following: $\sigma_{Q_p} = 0.1$, while $\sigma_{Q_v} = \sigma_{Q_\psi} = \sigma_{Q_h} = 0.5$.

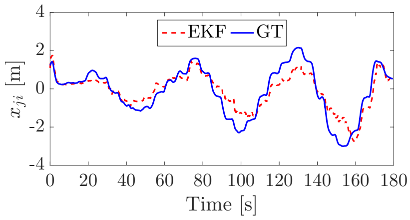
E. Preliminary Relative Localization Results

We performed flights with a Ladybird MAV around the fixed Bluetooth W1049B antenna in order to obtain preliminary insights of the performance of the EKF during flight. This information is used to design the collision avoidance strategy proposed in Section III. An Optitrack motion-capture system [53] was used to guide the MAV and record ground-truth 3D position, velocity, and orientation. Bluetooth RSSI data was recorded from the communication between the Ladybird MAV and the antenna. All data was recorded together at a rate of $5Hz$; this is a current limitation of the Bluetooth communication set-up discussed in Section IV.

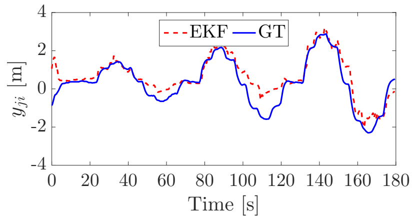
The data gathered was used to process the EKF off-board. The recorded velocity and orientation data from Optitrack was altered with Gaussian noise and used as measurements for the EKF. This simulated the measurement of these



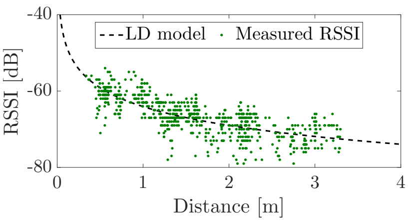
(a) Estimated range (red, dashed) between Ladybird MAV and Bluetooth antenna compared to ground truth (blue, solid) and estimate from reversing the LD model (green, dotted).



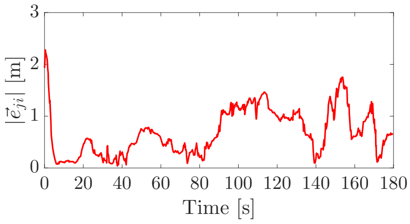
(b) Estimated (red, dashed) relative location of the Ladybird MAV along the x_B-axis of the antenna compared to ground truth (blue, solid).



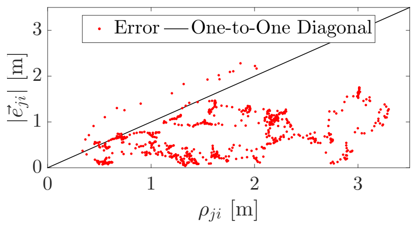
(c) Estimated (red, dashed) relative location of the Ladybird MAV along the y_B-axis of the antenna compared to ground truth (blue, solid).



(d) Measured RSSI (green, dotted) at different ranges and the LD model used (black, dashed). The parameters of the LD model are: $P_n = -63$ and $\gamma_l = 2.0$.



(e) Magnitude of pose error over time. Notice the convergence of the error in the initial seconds.



(f) Positive correlation between magnitude of pose error and distance below a diagonal line. The errors seen above the diagonal line are from the initial seconds prior to convergence.

Fig. 3: Preliminary localization results based on circular flights of a Bluetooth equipped Lisa-S Ladybird MAV around a fixed antenna. These results have been averaged over 50 iterations of artificial noise added to the velocity, height, and orientation measurements.

values using on-board sensors. The standard deviation of the noise given to velocity measurements is $\sigma_v = 0.2m/s$. The standard deviation of the noise given to altitude measurements is $\sigma_h = 0.2m$. The standard deviation of the noise given to orientation measurements is $\sigma_\psi = 0.2rad$. These values were also included in the measurement noise matrix $\mathbf{R}$. In the LD model of the EKF: $P_n = -63dB$ and $\gamma_l = 2.0$, assuming free-space propagation.

Figure 3 shows the results of the relative localization estimates achieved by the EKF. The antenna is $\mathcal{R}_i$ and it is trying to localize the MAV ($\mathcal{R}_j$). The top row shows the output of the EKF against Ground-Truth (GT) data. An immediate benefit observed is the significant reduction in error for the observed range, see Figure 3a. Estimates for x_{ji} and y_{ji} are shown in Figure 3b and Figure 3c. Let $\vec{e}_{ji}$ be the error in pose between the estimated position of $\mathcal{R}_j$ and the real position of $\mathcal{R}_j$. $\vec{e}_{ji}$ is expressed in $\mathcal{F}_{B_i}$. For a visual representation, see Figure 4. $|\vec{e}_{ji}|$, the magnitude of $\vec{e}_{ji}$, is shown over time in Figure 3e. Of particular interest for the subsequent development of a collision avoidance strategy is the observed increase of this error with the distance, as seen in Figure 3f. The increase is explained by the logarithmic relationship between RSSI and distance. The diagonal (one-to-one) line in Figure 3f indicates a maximum accepted error magnitude. When $|\vec{e}_{ji}|$ is above the line, then the error encompasses the position of $\mathcal{R}_i$ itself, rendering it insufficient to select an appropriate maneuver for collision avoidance. In the results, this is only observed for data over the first few seconds of flight, prior to the convergence of the EKF.

III. COLLISION AVOIDANCE BEHAVIOR

Relative localization between MAVs enables a series of team behaviors. The focus case-study in this paper is collision avoidance when operating in a confined space. This receives primary attention due to its severity as a failure condition.

This section describes the devised planar collision avoidance algorithm. It uses an altered version of the Velocity Obstacle (VO) [54] frame-work that is adjusted to suit the errors and short-comings of the relative localization algorithm. Lateral avoidance is preferred over height separation as a collision avoidance strategy in order to limit aerodynamic disturbances between MAVs [55] [16].

A. General Avoidance Strategy

A Collision Cone (CC) is a set of all velocities of an agent that are expected to lead to a collision with an obstacle [54]. Collision cones are so called because they are geometrically cone-shaped. Consider once more MAVs $\mathcal{R}_i$ and $\mathcal{R}_j$. We can then define a set CC_{ji} that includes all velocities of $\mathcal{R}_i$, defined in $\mathcal{F}_{B_i}$, which could lead to a collision with $\mathcal{R}_j$. See Figure 4 for a depiction. $\alpha_{CC_{ji}}$ is the *expansion angle* of the cone. It is expressed in radians and subject to $0 < \alpha_{CC_{ji}} < \pi$. The symmetry line of the cone is centered around the estimated bearing to the obstacle $\mathcal{R}_j$. The entire cone is then translated by the estimated velocity of $\mathcal{R}_j$ (available directly in $\mathcal{F}_{B_i}$ as $\dot{\vec{p}}_{jRi}$). Equation (12) and Equation (13) summarize how to determine CC_{ji} in $\mathcal{F}_{B_i}$ for a known $\alpha_{CC_{ji}}$, where x and y are points on x_{B_i} and y_{B_i}, respectively.

$$CC_{ji} = \{(x, y) \in \mathbb{R}^2; \alpha \in \mathbb{R}; |\alpha| \leq \frac{|\alpha_{CC_{ji}}|}{2} : \tan(\alpha)x = y\} \tag{12}$$

$$CC_{ji} \leftarrow \left(\mathbf{R}(\bar{\beta}_{ji}) \cdot CC_{ji}\right) \oplus \overline{\dot{\vec{p}}_{jRi}} \tag{13}$$

α is an angle in radians. All data required in Equation (13) is found in the output of the EKF proposed in Section II. $\bar{\beta}_{ji}$ is the estimated β_{ji}. $\overline{\dot{\vec{p}}_{jRi}}$ is the estimated $\dot{\vec{p}}_{jRi}$. When exact values of β_{ji} and $\dot{\vec{p}}_{jRi}$ are available, $\alpha_{CC_{ji}}$ is only dependent on the radii of the two MAVs (modeled as circular objects) [32]. Errors may be accounted for by further increasing $\alpha_{CC_{ji}}$ [20]. In the following, we propose a method to establish the expansion angle tailored to the errors of the Bluetooth relative localization scheme.

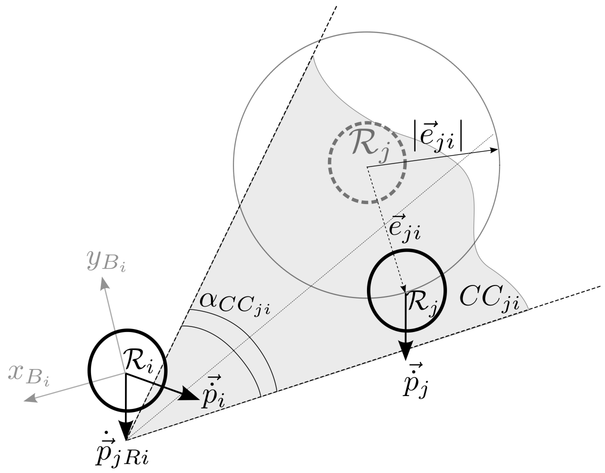


Fig. 4: Depiction of CC_{ji} that $\mathcal{R}_i$ holds with respect to $\mathcal{R}_j$. The dashed circle is the estimated location of $\mathcal{R}_j$. $\vec{e}_{ji}$ is the localization error.

In Figure 3f it is observed that the magnitude of the localization error increases with the distance, extrapolated to the following relationship:

$$E(|\vec{e}_{ji}|) = \frac{1}{\kappa_\alpha} \cdot \bar{\rho}_{ji}, \tag{14}$$

where κ_α is a constant coefficient describing the quality of the estimate. $E(\cdot)$ is the expected value. $\bar{\rho}_{ji}$ is the estimated range between $\mathcal{R}_i$ and $\mathcal{R}_j$. Note that if $\kappa_\alpha < 1$ then $E(|\vec{e}_{ji}|) > \bar{\rho}_{ji}$, meaning that the potential bearing estimation error is 2π and it does not provide useful information for collision avoidance. If $\kappa_\alpha \geq 1$ then the estimate is sufficient to select a collision escape trajectory (in Figure 3f we observe that the worst case error is slightly below the diagonal line of $\kappa_\alpha = 1$). Based on this knowledge, we define the expansion angle $\alpha_{CC_{ji}}$ based on the implication of $E(|\vec{e}_{ji}|)$ on the bearing error, as in Equation (15).

$$\alpha_{CC_{ji}} = 2 \cdot \tan^{-1}\left(\frac{\bar{\rho}_{ji} + r_i + r_j + \varepsilon_\alpha}{\kappa_\alpha \cdot \bar{\rho}_{ji}}\right) \tag{15}$$

r_i and r_j are the radii of the MAVs. In a homogeneous team: $r_i = r_j$. The factor κ_α dictates the lower limit asymptote of $\alpha_{CC_{ji}}$ as $\bar{\rho}_{ji} \rightarrow \infty$. Its impact may be appreciated in Figure 5. The asymptote ($\alpha_{CC_{asymptote}}$) is determined using Equation (16).

$$\alpha_{CC_{asymptote}} = \lim_{\bar{\rho}_{ji} \rightarrow \infty} \alpha_{CC_{ji}} = 2 \cdot \tan^{-1}\left(\frac{1}{\kappa_\alpha}\right) \tag{16}$$

Based on the worst case scenario for relative localization error, i.e. $\kappa_\alpha = 1$, the asymptotic angle is $\pi/2$. ε_α is an additional margin designed to adapt the behavior of the MAVs depending on the (estimated) size of the confined space in which they move. A method for the appropriate selection of ε_α is discussed in more detail in Section III-B.

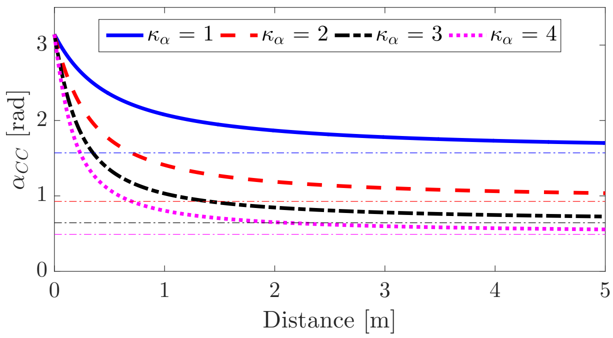


Fig. 5: Effect of κ_α on α_{CC} along distance ρ_{ji}. The other parameters are set to $r_i = r_j = 0.1m$ and $\varepsilon_\alpha = 0.5$. The straight lines represent the relevant asymptotes.

In a team of m MAVs, each member $\mathcal{R}_i$ holds $m - 1$ collision cones that it can superimpose into a single set CC_i:

$$CC_i = \bigcup_{j=1}^{m-1} CC_{ji} \tag{17}$$

If, during flight, $\dot{\vec{p}}_i \in CC_i$, then a *clock-wise* search about the z_{B_i} axis (starting with the current direction of flight) is used to determine the desired velocity for escape from a collision course. The clock-wise search aims to hold the nominal desired magnitude for $\dot{\vec{p}}_i$. If no solution is found, then the search is repeated with incremented speed.

B. Preserving Behavior in Different Room Sizes

Equation (15) does not generalize well to environments of different sizes when all its parameters (r_i, r_j, κ_α, ε_α) remain constant. Too small values of κ_α and/or too large values of ε_α can enlarge the collision cone too much and restrict freedom of movement when operating in smaller rooms. This brings two separate disadvantages, both in part culprits for eventual collisions:

1) Oscillations/instability in MAV trajectories.
2) Convergence of the EKF suffers due to small noise-like movements.

Appropriate scaling of the collision cone is achieved by altering ε_α, which dictates the slope for the change in α_{CC} at smaller distances, see Figure 6.

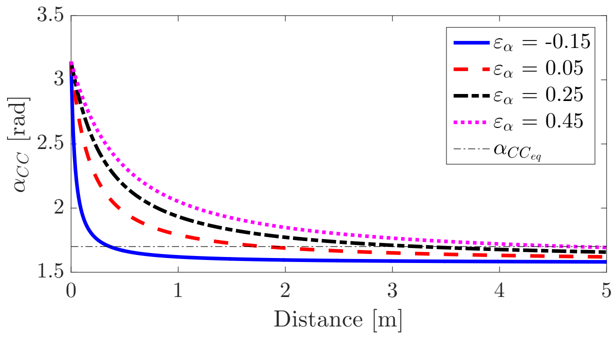


Fig. 6: Effect of ε_α on α_{CC} along ρ_{ji}. The other parameters are set to $r_i = r_j = 0.1m$ and $\kappa_\alpha = 0.5$.

By re-arranging Equation (15), ε_α can be determined with the following rule:

$$\varepsilon_\alpha = \kappa_\alpha \cdot \rho_{eq} \cdot \tan\left(\frac{\alpha_{CC_{eq}}}{2}\right) - (r_i + r_j) - \rho_{eq}, \qquad (18)$$

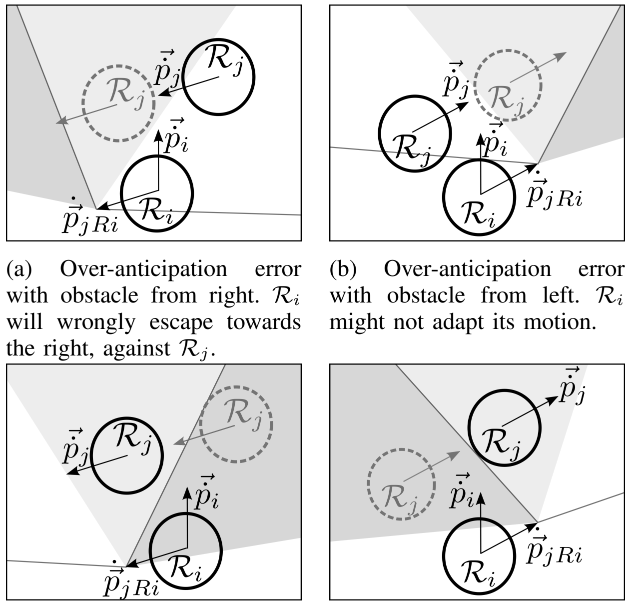
This equation relies on the pair of parameters ρ_{eq} and $\alpha_{CC_{eq}}$. $\alpha_{CC_{eq}}$ is the desired angle of expansion at a distance ρ_{eq}. For a given κ_α, ε_α can be adjusted with Equation (18) to adapt the expansion of the cone when ρ_{eq} changes. Note that $\alpha_{CC_{eq}} > \alpha_{CC_{asympote}}$. Equation (18) sets the limit: $\varepsilon_\alpha > -(r_i + r_j)$.

The selection of ρ_{eq} and $\alpha_{CC_{eq}}$ is left to the designer based on the expected circumstances. Due to the conservative choice of κ_α, lower values of ρ_{eq} would be preferred to enable mobility. In all tests in this article, ρ_{eq} is at a distance that is half of the expected side length of the (square) arena. $\alpha_{CC_{eq}}$ is kept at a $1.7rad$. In a realistic adaptive task, under the assumption that the MAVs are equipped with a wall-sensor, then they could define ρ_{eq} on-board based on the distance to the surrounding walls.

C. Correction Against Chattering and Motion Anticipation

The use of collision cones based on the EKF outputs suffers from two issues, discussed below.

- *Over-anticipation or trailing of position estimate.* Localization estimates may include a bias for over-anticipation or trailing with respect to the ground-truth position. Over-anticipation is when the estimate is in front of the actual position, as if leading it. Trailing is when the estimate is behind the actual position, as if following it. The error is stochastically dependent on the noise realizations and disturbances present at a given time step. For head-on and rear-end collision avoidance, this is not an issue because the collision cone remains approximately centered around the correct relative bearing. For side collision avoidance (collisions coming from oblique angles with respect to motion), the rotation and shift of the collision cone make anticipation unfavorable. It is responsible for a failure case where the MAV escapes to the wrong side or fails to react. This is visualized in Figure 7a and Figure 7b. Alternatively, trailing errors have been observed to be less problematic because the collision cones still tend to point and cover the correct direction. See Figure 7c and Figure 7d, where the collision cone still prompts a valid escape maneuver.
- *Chattering of the collision cones.* Due to the noise in the EKF outputs, the collision cones are subject to chattering in both their orientation (due to noise in $\vec{p}_{ji}$), and in their translation (due to noise in $\dot{\vec{p}}_{jRi}$).



(a) Over-anticipation error with obstacle from right. $\mathcal{R}_i$ will wrongly escape towards the right, against $\mathcal{R}_j$.

(b) Over-anticipation error with obstacle from left. $\mathcal{R}_i$ might not adapt its motion.

(c) Trailing error with obstacle from right. $\mathcal{R}_i$ will escape towards bottom right.

(d) Trailing error with obstacle from left. $\mathcal{R}_i$ will escape towards right.

Fig. 7: Illustration of over-anticipation and trailing errors for a collision from right (top) and left (bottom) at approximately oblique collision angles. The gray collision cone is the one based on the EKF output. The white collision cone is the one based on the real position.

The issues were tackled together by implementing a Moving Average Filter (MAF) on the EKF output estimates of $\vec{p}_{ji}$ and $\dot{\vec{p}}_{jRi}$. The MAF inherently introduces a trailing error in the position estimate that corrects over-anticipation errors. The MAF also increases the smoothness of the escape trajectory due to lower chattering of the collision cone. Note that the MAF is only used to calculate the collision cones and is applied in a separate process from the EKF. Combining the two would promote divergence in the EKF's output due to an un-modeled non-linearity introduced in the system. It should also be noted that the use of the MAF should be just enough to decrease chattering and encourage a minor trailing error. In general, it is applied over the last few time steps.

D. Connection with Velocity Obstacle Methods

One may note the resemblance of the proposed avoidance strategy to the VO method. The difference is that VO selects a new flight direction that minimizes the required change in velocity [32] [33] as opposed to the clockwise search suggested here. VO is notoriously prone to *reciprocal dances*

[56]. These are oscillations in the trajectory when entities heading towards each-other repeatedly select the same escape direction, leading to a left-right "dance". Reciprocal dances spawn when each entity (wrongly) assumes that the other will not change its course and avoidance will not be reciprocal. Several variants attempt to solve this issue by making the opposite assumption, i.e. that both entities will try to evade the collision. Examples include Reciprocal Velocity Obstacle (RVO) [57], Hybrid Reciprocal Velocity Obstacle (HRVO) [58] [59], and Optimal Reciprocal Collision Avoidance (ORCA) [60]. In this case, however, due to the potential for large relative localization errors, the MAVs are not made to assume that the others will participate in a suitable and reciprocal escape maneuver. Reciprocal variants of VO are thus discouraged and reliance on own estimates and avoidance is preferred. The clockwise search encourages a preference for right-sided maneuvers with respect to the current flight direction, automatically resolving reciprocal dances.

IV. TEST SET-UP

This section describes the tests that have been set up to establish the performance of the combined system in a realistic team flight. This has first been done in simulation in order to establish the limitations, and later in the real-world, with the different objective of establishing the reality gap resulting from the use of real RSSI measurements and real on-board velocity estimates for data exchange between MAVs.

A. Description of Arbitrary Task for Performance Testing

A controller is designed to instantiate an arbitrary task and applied homogeneously to all MAVs. The task is designed such that the MAVs repeatedly seek to pass *through* the center of the arena. This is made so as to provoke several random potential collision scenarios and observe if/how these scenarios are resolved.

Consider a team of m homogeneous MAVs. Each MAV $\mathcal{R}_i$ is controlled in velocity. Let $\dot{\vec{p}}_{i_{cmd,k}}$ be the desired velocity for $\mathcal{R}_i$ expressed in its body-frame $\mathcal{F}_{B_i}$ at a given time-step k. Let d_{wall_i} be the distance between $\mathcal{R}_i$ and the arena border that is closest to it, with d_{safe} being a safety distance to the arena's borders. Remember that each robot $\mathcal{R}_i$ features $m-1$ EKF instances to keep track of the other members and uses their outputs to determine its collision cone set CC_i, see Equation (17). At each-time step k, the EKF outputs are updated and CC_i is re-calculated. $\dot{\vec{p}}_{i_{cmd,k}}$ is then chosen as follows: $\dot{\vec{p}}_{i_{cmd,k}} = \dot{\vec{p}}_{i_{cmd,k-1}}$ unless conditions M1 and M2 take place.

M1: $d_{wall_i} < d_{safe}$ and $\dot{d}_{wall_i} < 0$. This means that $\mathcal{R}_i$ is close to the arena border and approaching it. Then, $\dot{\vec{p}}_{i_{cmd,k}}$ is rotated towards the center of the arena. See Figure 8.

M2: $\dot{\vec{p}}_i \in CC_i$. This means that the current velocity of $\mathcal{R}_i$ could lead to a collision with one or more team members. An escape velocity is sought according to the strategy proposed in Section III.

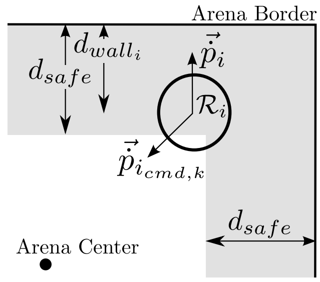


Fig. 8: Depiction of MAV $\mathcal{R}_i$ subject to condition M1 at a time step k. $\mathcal{R}_i$ is closer to the arena border than d_{safe} allows, and moving towards it (see its velocity, $\dot{\vec{p}}_i$). When this happens, the commanded velocity ($\dot{\vec{p}}_{i_{cmd,k}}$) is towards the center of the arena.

Note that M1 supersedes M2 to ensure that the MAVs remain within the confines of the arena. At all time-steps, unless other-wise commanded by the collision avoidance algorithm, $|\dot{\vec{p}}_{i_{cmd,k}}| = v_{nominal}$, where $v_{nominal}$ is a fixed speed magnitude. The MAVs fly at the same height at all times.

The controller described above is implemented as a call-back function upon the reception of new data from other MAVs. In the experiments this ran at approximately $5Hz$ due to the limitations of the real-world implementation (see Section IV-D). Furthermore, the MAVs always maintain the same heading with respect to North, purposely taking advantage of the 6-Degrees of Freedom (DOF) dynamics of quadrotors. In all experiments $v_{nominal} = 0.5m/s$. In all simulations $d_{safe} = 0.25m$. In all real-world tests, for conservative/safety reasons, d_{safe} was increased to $0.5m$.

Section IV-C and Section IV-D describe the implementation of the above in the simulation environment and the real-world, respectively. At this early research stage, it will be noticed that both implementations equally rely on an external position sensor in order to enforce M1. In a realistic task, arena borders would be detected using environment features such as walls (e.g. [61]) and the center of the arena would represent an attraction way-point.

In all instances, the MAVs begin the task at different corners of the arena (this is approximate for the real-world tests). The EKF is initialized such that the initial position estimate is towards their initial flight direction (i.e. approximately the center of the arena). All other states are initialized as null. The covariance matrix of the EKF is initialized as an identity matrix.

B. Testing Several Density Configurations

The performance of the task described in Section IV-A is dependent on how crowded/dense the airspace is. This has been investigated by altering both arena size and MAV

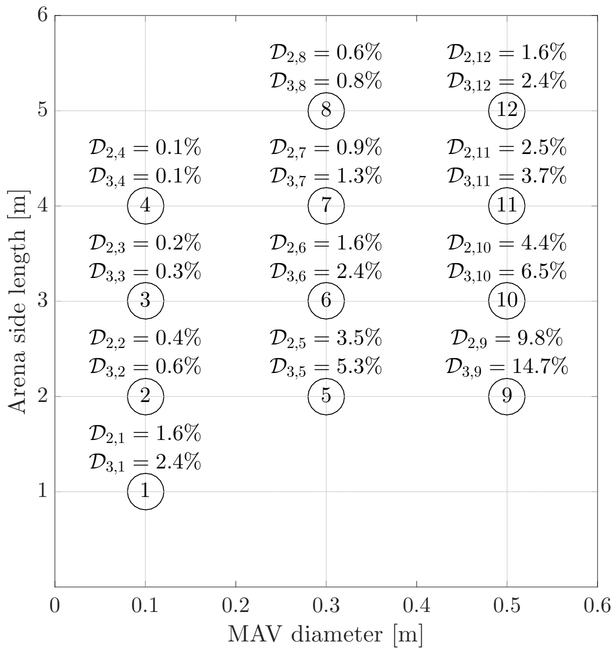


Fig. 9: Matrix graph of all tested configuration pairs between MAV diameter and side length of square arena. The configuration numbers as shown in the white circles are referenced throughout the remainder of this article. $\mathcal{D}_{m,c}$ is the airspace density for configuration c when featuring a team of m MAVs.

diameter in twelve different configurations. The investigated configurations and their respective densities are shown in Figure 9. These will be referred to throughout the remainder of this article by the numbers in the circles.

Density is calculated by modeling each MAV as a circle in a square arena. Let $\mathcal{D}_{m,c}$ denote the density for configuration c with m MAVs. It is calculated as in Equation (19).

$$\mathcal{D}_{m,c} = \frac{m \cdot \pi r_c^2}{s^2} \tag{19}$$

r_c is the radius of an arbitrary MAV at configuration c (all MAVs in a configuration are homogeneous). s_c is the side length of the squared arena at configuration c. All tests are performed with two MAVs ($m = 2$) and three MAVs ($m = 3$).

C. Simulation Environment Set-Up

The simulation environment was built using the Robotics Operating System (ROS) [62]. It adopts the *Gazebo* physics engine [63] and the *hector-quadrotor* [64] simulation which together provide a validated platform. [1] Multiple instances of a quad-rotor may be launched in a simulation run. The core functions (i.e. relative localization EKF and collision avoidance controller) are developed for and within Paparazzi Unmanned Air Vehicle (UAV) software [65] [66] [2]. This is so as to be readily portable to the real-world set-up described in Section IV-D. A ROS module (a.k.a. "node" [62]) for each MAV simulates the presence of a Bluetooth RSSI sensor and subsequently enforces the controller described in Section IV-A. The module runs at $5Hz$ to match the communication speed between MAVs achieved in the real-world (see Section IV-D).

The RSSI signal was simulated using the LD model ($P_n = -63dB$, $\gamma_l = 2.0$) with added Gaussian noise as well as horizontal lobes as a function of relative bearing. The standard deviation of the added noise, with the exception of the results discussed in Section VI-C, is $5dB$. The lobes were arbitrarily modeled using a third order Fourier series with unitary weights, see Figure 10a. With Figure 10b, we see that the achieved performance of the simulated on-board relative localization scheme features similar error magnitudes as observed in Figure 3e when a similar test is repeated in simulation.

Each configuration from Figure 9, unless otherwise stated, has been tested with 100 trials featuring a maximum trial time of $500s$. The simulations are interrupted whenever the actual distance between any two MAVs is smaller than the sum of their radii, indicating a collision. A screen-shot of a simulation with 3 MAVs is shown in Figure 10c.

D. Real-World Environment Set-Up

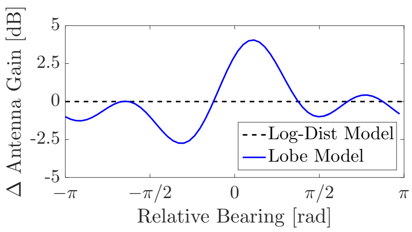
We executed real-world experiments using AR-Drones 2.0 [67] running Paparazzi. The drones all flew at $1.5m$ above the ground within a $4m \times 4m$ arena. Figure 11 shows a picture of an on-going experiment with 3 drones.

A BLED112 Bluetooth Smart USB Dongle was used to provide the AR-Drones with Bluetooth 4.0 (Low-Energy) capabilities [68]. All computations for relative localization and collision avoidance are run on-board of the AR-Drones. The LD model in the EKF filter was given: $P_n = -67dB$ and $\gamma_l = 2.0$. P_n was obtained by a brief hand-held calibration measurement. The choice of γ_l was based on the free-space assumption [46]. Communication between the AR-Drones was *direct* via Bluetooth. The data was sent and received by means of advertising messages scheduled using a Self-Organized Time Division Multiple Access (STDMA) algorithm [69]. Under the STDMA algorithm, each MAV's Bluetooth antenna alternates between advertising and listening, achieving data exchange at a rate of $\approx 5Hz$. This enabled direct communication circumventing the Master-Slave paradigm otherwise enforced by the Bluetooth standard [30].

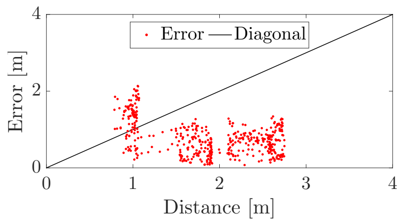
Two different real-world tests were performed. Test #1 explored the impact of using real RSSI measurements and communication. Test #2 then explored the effect of using

[1]At the time of writing: ROS is freely available at www.ros.org; Gazebo is freely available at www.gazebosim.org; Hector-quadrotor is freely available at wiki.ros.org/hector_quadrotor.

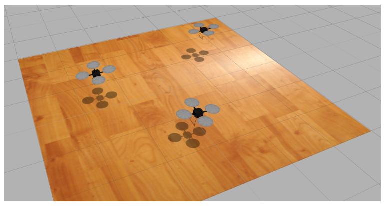
[2]Paparazzi UAV is an open-source UAV/MAV auto-pilot software available at https://github.com/paparazzi/paparazzi.



(a) Lobes applied to the simulated RSSI signals on each simulation. The lobes superimpose as elaborated in [10].



(b) Localization error over simulation of a concentric circular flight about a static antenna The higher errors at 1m are prior to the convergence of the EKF.



(c) Gazebo visualization of a simulation with 3 MAVs.

Fig. 10: Figures relating to the development of the Gazebo/ROS simulation environment.

on-board velocity estimates. The two tests are discussed in more detail in the next two paragraphs.

Test #1 (Optitrack-based state estimation): The primary objective of this test was to establish the performance of the relative localization and collision avoidance algorithm when using real RSSI measurements and Bluetooth communication. On-board state estimation of velocity, magnetic North orientation, and height was purposely avoided in order to be able to ensure regulated noise and isolate the impact of using real Bluetooth RSSI. Optitrack [53] was used to provide the MAVs with estimates of their *own* states via a Wi-Fi link. These were altered the same Gaussian noises $\sigma_v = 0.2m/s$ and $\sigma_\psi = 0.2rad$ upon being entered into the EKF. Furthermore, Optitrack position data was used to guide MAVs according to condition M1 as proposed in Section IV-A. The enforced arena size in all experiments was $4m \times 4m$. These experiments are approximately analogous to Configuration 11 from the simulated tests (AR-Drones 2.0 are slightly larger in diameter than $0.5m$). Experiments were performed with both two AR-Drones and three AR-Drones. For flights with two AR-Drones, the MAF averaged over the last 3 time-steps. For flights with three AR-Drones, the MAF averaged with the last time-step.

Test #2 (On-board velocity estimation): These tests were performed using two AR-Drones. The objective was to determine the impact of using realistic velocity sensors on the relative localization performance and its repercussions on collision avoidance. Instead of relying on Optitrack, the AR-Drones estimated their own velocity using on-board sensors and the the Optical Flow module available within Paparazzi [3]. The on-board estimates were then directly communicated between the AR-Drones using Bluetooth. For safety reasons, and to isolate the impact of the relative localization estimate on collision avoidance, on-board velocity estimates were *only* used as inputs for the EKF relative localization. The velocity controller of the AR-Drones remained reliant on Optitrack. This ensured controlled flight within the confines of the arena. The drones kept a constant heading towards North and the same height at all times. No noise was artificially added to these measurements. All other parameters remained as in Test #1.

[3]See the module *computer_vision/opticflow* for a more in-depth description

Fig. 11: Picture during a real-world experiment with 3 AR-Drones 2.0 (one left, one middle, and one right) inside the arena.

V. Results

A. Simulation Results

All configurations presented in Section IV-B have been tested under the simulation environment described in Section IV-C. The objective of the simulations is to study the performance trends under different environments and the limitations of the system. The parameter used to assess the performance is the *flight-time to collision*, which is the time that the MAVs managed to fly within the arena whilst avoiding collisions. The mean flight-time to collision for each configuration is shown in Figure 12. Remember that simulations were stopped after $500s$ of flight in the event of no collisions. For all configurations, flights with three MAVs show a lower performance with respect to two MAVs. In the simulations, the introduction of an additional MAV does not affect the relative localization performance between MAVs. Therefore, the performance drop is a result of the team dynamics at play, namely: increased airspace density, and decreased freedom of movement due to superposition of collision cones. These two factors are analyzed in this section.

When the arena side length remains constant and the MAV diameter increases, a decrease in mean flight-time is

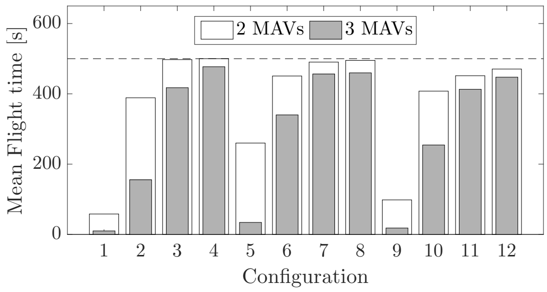
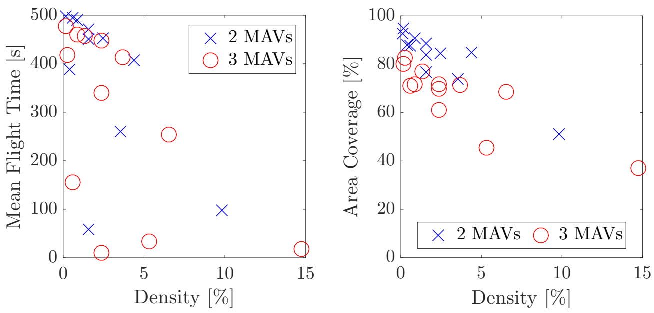


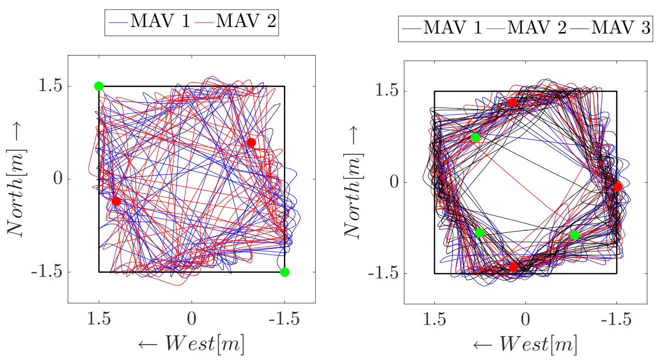
Fig. 12: Mean flight-time to collision for all configurations with active collision avoidance. The maximum simulation time for each trial was $500s$ in the event of no collisions. The mean flight times without collision avoidance, not seen in this figure, range between $3.9s$ and $14.3s$.



(a) Mean flight-time with respect to density

(b) Mean area-coverage with respect to density

Fig. 13: Flight parameters with respect to airspace density based on simulation results.



(a) Emergent circular trajectory with two MAVs.

(b) Emergent circular trajectory with three MAVs.

Fig. 14: Trajectories from two exemplary simulated flights of $500s$ extracted from configuration 10 showing the emergent circular behavior with two MAVs (left) and three MAVs (right). The starting positions are shown in green (note that for the flight with three MAVs this was actually at the corners, but the first few time-steps were not logged). The final positions are shown in red.

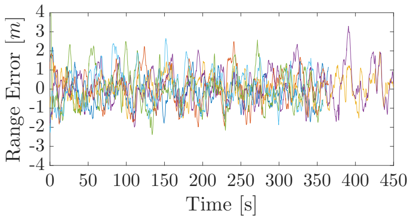
systematically present. This is observed when comparing within the configuration triads 4-7-11, 3-6-10, and 2-5-9, and the pair 8-12. The result is analogous when MAVs of the same diameter are used in arenas of different sizes, as may be noticed by observing the configuration quartets 1-2-3-4, 5-6-7-8, and 9-10-11-12. This implies that a lower density improved the probability of success, but this is found to not strictly be the case. Figure 13a shows the flight time to collision as a function of the airspace density. A portion of configurations show low results in spite of the low airspace density, and are outliers in the negative linear trend. These correspond to configurations 1, 2, 5, and 9, which feature smaller arena sizes. The conclusion is that room size affects performance even when airspace density remains constant. This is a limitation of the current status of the system when operating in smaller room sizes. Its causes are discussed in Section VI-B.

Figure 13b shows the impact of airspace density on area coverage for all flights with two MAVs and three MAVs. Area coverage is measured as follows. The arena is divided in sections. A section is then considered covered if at least one of the MAVs crosses it during a trial. Area coverage is the ratio of covered sections to the total number of sections. The calculation was performed using standardized sections of $0.20m \times 0.20m$. Two patterns are discerned. The first is the general trend that a higher airspace density leads to a lower overall coverage. This is a combined effect of a) lower flight times, providing less opportunity for movement, and b) decreased freedom of movement due to larger portions of the arena being covered by collision cones. The second pattern is that flights with three MAVs systematically achieve lower area coverage if compared to flights with two MAVs at the same density. This is explained by analyzing the flight trajectories in more detail, which show an emergent circular behavior. This behavior may be appreciated in Figure 14, showing two exemplary runs from a simulation with two (Figure 14a) and three (Figure 14b) MAVs from configuration 10. When more than one MAV to avoid is present, the superposition of multiple collision cones significantly discourages the pursuit of the desired trajectory. The result is clock-wise motion along the sides of the arena for all MAVs. Oscillations along the border are observed as conditions M1 and M2 alternate.

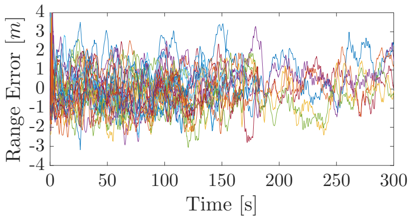
B. Real-World Results with Optitrack-based State Estimation (Test #1)

Four flights were performed with two AR-Drones in a $4m \times 4m$ arena. The cumulative flight-time was $25.3min$. In this time, the MAVs only suffered from one collision, which took place in the second flight after $5.6min$. The other flights lasted $6.1min$, $7.6min$, and $6.0min$; they were ended manually in order to preserve battery health in light of low battery voltage.

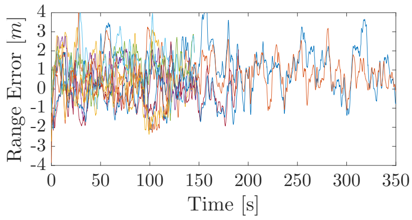
Six flights were performed with three ARDrones for a cumulative time of $15.3min$. Five out of six flights ended due to collisions. The flights ending with collisions reached a mean flight time of $160s$ ($2.7min$) before ending with collisions. The shortest flight was $33s$, the longest flight was $5.2min$. The other flights lasted $1.9min$, $2.6min$, and



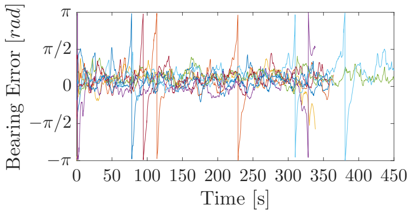
(a) Overview of range estimate error during all flights with two AR-Drones with Optitrack-based state estimation (Test #1). The RMSE is $0.86m$.



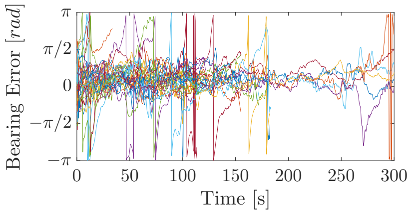
(b) Overview of range estimate errors during all flights with three AR-Drones with Optitrack-based state estimation (Test #1). The RMSE is $1.14m$.



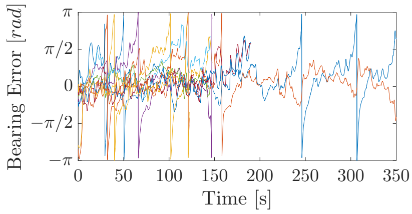
(c) Overview of range estimate errors during all flights with two AR-Drones with on-board velocity estimation (Test #2). The RMSE is $1.18m$.



(d) Overview of bearing estimate errors during all flights with two AR-Drones with Optitrack-based state estimation (Test #1). The RMSE is $0.57rad$.

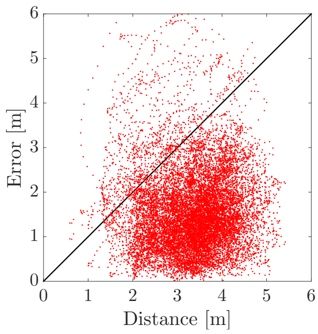


(e) Overview of bearing estimate errors during all flights with three AR-Drones with Optitrack-based state estimation (Test #1). The RMSE is $0.70rad$.

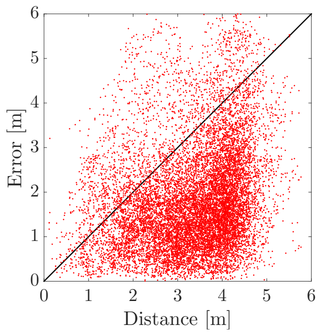


(f) Overview of bearing estimate errors during all flights with two AR-Drones with on-board velocity estimation (Test #2). The RMSE is $0.77rad$.

Fig. 16: Overview of all relative range (top) and relative bearing (bottom) errors. All plots on the left relate to flights with two AR-Drones with Optitrack-based state estimation (Test #1). All plots in the middle relate to flights with three AR-Drones with Optitrack-based state estimation (Test #1). All plots on the right relate to flights with two AR-Drones with on-board velocity estimation (Test #2).



(a) Localization error magnitude for all flights two AR-Drones in Test #1.



(b) Localization error magnitude for all flights three AR-Drones in Test #1.

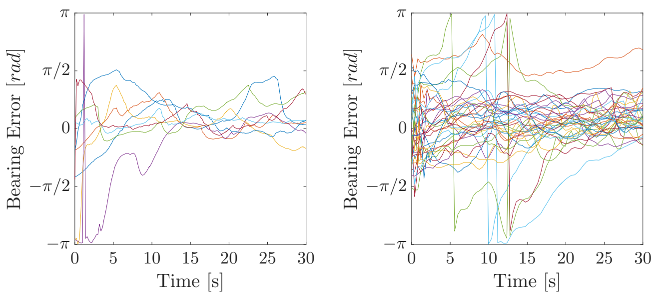
Fig. 15: Overview of magnitude of relative position estimate error ($|\vec{e}_{ji}|$ between all MAVs during all flights with two AR-Drones (left) and three AR-Drones (right). All data shown here equally relied on off-board self-state estimates (Test #1).

$3.0min$. The flight without a collision was manually ended after $2.0min$ due to low battery voltage. Under these results it may be said that a system with three MAVs can expect a collision once every $184s$ ($\approx 3min$) of flight under the proposed task.

Figure 15 shows the magnitude of the localization error ($|e_{ji}|$) for all combinations and during all flights together with two AR-Drones (Figure 15a) and three AR-Drones (Figure 15b) in Test #1. For all flights with two AR-Drones, 92% of all estimates are below the expected line of $\kappa_\alpha = 1$. The minimum performance by an MAV over a flight is 84.8%, and the maximum is 97.6%. For all flights with three AR-Drones, 84% of points are below the expected line.

Figure 16 shows the errors in bearing and range for all relative estimates during all flights. For flights from Test #1: the range error is shown in Figure 16a and Figure 16b, and the bearing error is shown Figure 16d and Figure 16e. The mean error with two MAVs features a Root Mean Squared Error (RMSE) of $0.57rad$ for bearing estimates and $0.86m$ for range estimates. With three MAVs, the RMSE rises to $0.70rad$ and $1.14m$ for bearing and range estimates, respectively.

Of particular interest is the amount of times that the error temporarily diverges towards $\pm\pi$. In spite of the shorter cumulative flight time, this error case is more frequent in the flight with three MAVs. The error does not necessarily lead to collisions in light of the homogeneous application of the controller to all MAVs and the abstinence from assuming reciprocity in the collision avoidance. Nevertheless, it does introduce a temporary uncertainty in the system that is not accounted for by the collision avoidance. Furthermore, the convergence rate for bearing estimates over flights with three AR-Drones is worse than with two AR-Drones. This may be appreciated in Figure 17, which shows the first $30s$ of Figure 16d and Figure 16e in more detail. Convergence times for flights with three MAVs reach up to $30s$ prior to settling (Figure 17b). By comparison, the convergence in flights with two AR-Drones only (Figure 17a) is found to be at most within $5-10s$.



(a) Bearing error for all flights with two AR-Drones in the first $30s$.

(b) Bearing error for all flights with three AR-Drones in the first $30s$.

Fig. 17: Comparison of bearing estimate errors between all performed flights with two AR-Drones (left) and three AR-Drones (right) in the first 30 seconds of flight. All data shown here equally relied on off-board self-state estimates.

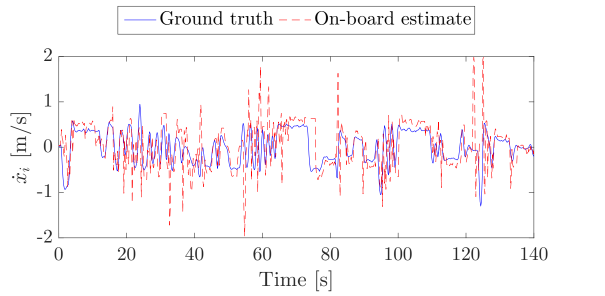


Fig. 18: Velocity estimation (red, dashed) of an AR-Drone along the x_B axis against ground truth velocity (blue, solid). Notice the small spikes between $30s$ and $40s$ and the larger spikes at $\approx 55s$ and $\approx 125s$. Furthermore, notice the occasional over/under estimation in the regions between $70s$ to $80s$ and $100s$ to $110s$.

C. Real-World Results with On-Board Velocity Estimation (Test #2)

Five flights were performed with two AR-Drones for a cumulative time of $16min$. All flights ended in collisions. The mean time to collision was $192.4s$ ($3.2min$). The longest recorded flight time was $6.0min$. The shortest flight was $2.0min$. The other flights were $2.4min$, $2.4min$, and $3.1min$. This is a significant drop in performance if compared to the flights from Test #1 with two AR-Drones. The reason is in the increased error in the relative range and bearing estimates when compared to flights with two MAVs using Optitrack velocity estimates, see Figure 16c and Figure 16f. The RMSE of the relative range estimates was $1.18m$, whereas the RMSE for relative bearing estimates was $0.77rad$.

The loss in relative localization accuracy is attributed to the disturbances present in the on-board velocity estimates. Figure 18 shows an exemplary estimate of the velocity along the x_{B_i} axis by one AR-Drone during a flight. At different intervals, it may be seen that the velocity was over or under estimated for an extended period of time. Furthermore, significant spikes can be seen at $\approx 55s$ and $\approx 125s$. These spikes were manually limited to a maximum magnitude of $2m/s$, yet impose significant disturbance in the localization estimate. Finally, the standard deviation of the error reaches $\approx 0.4m/s$. This was not accounted for in the EKF, which still assumed $0.2m/s$.

VI. DISCUSSION

A. Performance of Relative Localization

The flights with 2 AR-Drones from Test #1 returned low relative localization errors and successful collision avoidance over a prolonged flight time of $25min$ with only one collision. However, a noticeable loss in relative localization performance was measured when introducing a 3^{rd} MAV. The effects were longer convergence times as well as higher relative bearing/range errors that negatively impacted the performance of the collision avoidance system when compared to the simulation results. The trend is assumed to get worse with team larger than 3. A similar decrease in performance was also observed when using on-board velocity estimates. This was due to a combination of over-under estimation of velocity or occasional spikes in the measurements.

The relative localization scheme uses an EKF. This may be criticized for its reliance on a Gaussian noise driven model, which fails to provide robustness against un-modeled disturbances. Other methods such as robust [70] or adaptive [71] variants of Kalman filters, or Particle Filters (PFs) [48], could be be better suited to deal with the circumstances. However, a naive change in filter can bring increased cost in computational resources without necessarily guaranteeing a higher quality output. This is because there are a number of other limitations. One is that the logarithmic decrease in RSSI is intrinsically insufficient to measure changes in range at larger distances. Another limitation stems from the proposed process update equation (see Equation (7)). In order to provide a general scheme and abstain from introducing more complexities in the system, it is based on the null assumption that all velocities remain constant. Improvements may come from including more complex dynamic properties in the process equation, i.e. acceleration and jerk.

To make the case that a change in filter is not necessarily to be associated with an improved performance, we compare the performance of the EKF to that of the Unscented Kalman Filter (UKF). The UKF is correct to a higher order [72] and does not need to be influenced by the assumption of Gaussian noise [73]. Two implementations of the UKF are used, one with distribution parameter of 2 (denoted UKF_2) and one with a distribution parameter of 0 (denoted UKF_0). UKF_2 incorporates Gaussian noise, whereas UKF_0 abstains from an initial a-priori knowledge [73]. Figure 19 shows the results for the same preliminary trial run previously discussed in Section II-E. All filters were applied to the same realization of artificial noise on the measurements and

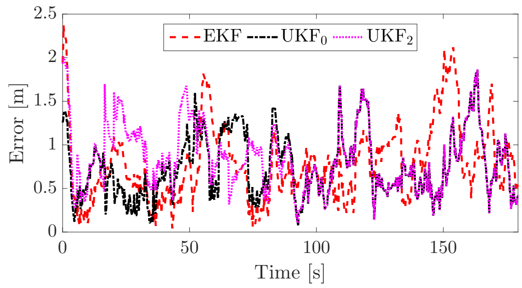


Fig. 19: Comparison of localization error ($|e_{ji}|$) with EKF (red,dashed), Gaussian UKF (purple, dotted), and non-Gaussian UKF with distribution parameter = 0 (black, dash-dotted) against ground truth data (blue, solid). The results are from one realization of artificial noise on the measurements from the same data-set used in Figure 3.

featured the same initial conditions. It may be seen that the performance is comparable. One reason is that the UKF's main strength (lack of linearization [74]) is in vain due to the low non-linearity of the process/measurement equations [75]. This may change if the process equation is altered as previously suggested. Another reason is that there is still a considerable impact from un-modeled disturbances in the environment.

Further investigations are encouraged in order to define a filter that can lower the expected worst-case error. This would benefit the system as a whole. The collision cone parameter κ_α could be reduced without introducing additional risk. This discussion is continued in Section VI-B.

B. Performance of Collision Avoidance

In simulation, all configurations have also been tested *without* active collision avoidance. In this case, the MAVs are only subject to condition M1. The obtained mean flight times range between $3.9s$ and $14.3s$. A z-test with 95% confidence level [76] shows a statistically significant improvement in flight time for all configurations.

Figure 13a shows that an increase in airspace density is directly correlated with a decrease in performance. Smaller rooms show poorer performance than larger rooms despite similar density. The parameter ε_α, as explained in Section III-B, implements room scaling within the collision cones. However, performance cannot reach the same levels unless the other relevant parameters (i.e. call-back rate ($5Hz$), $v_{nominal}$, d_{safe}, sensor noises) are also changed accordingly. Two reasons for this are:

- *The ratio of arena size to $v_{nominal}$ decreases in smaller rooms*. The rate of data exchange is $5Hz$, and this limits the decision rate of the collision avoidance controller. With other control parameters remaining constant, the relative distance traveled in smaller rooms is higher than in larger spaces, prompting a more chaotic behavior.
- *Maneuver selection*. In smaller rooms, M1 has higher chances of being called due to more frequent proximity to the arena borders.

Collisions during real-world flights with three MAVs occurred along the edges of the arena. This is also observed in simulation. For configuration 11, 81% of the collided simulated flights with three MAVs ended within $0.5m$ of the arena borders. By comparison, only 35% of collided flights with two MAVs ended within this space. An example is shown in Figure 20 from a real flight, recounted by the three events below.

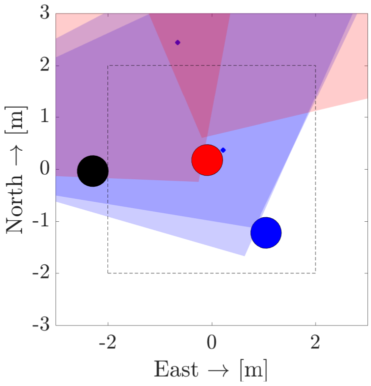
1. One AR-Drone ends up "trapped" along the boundaries and reluctant to make movements towards the center for fear of collisions. In Figure 20a we see that the bottom right AR-Drone (blue) turns towards the right.
2. Another MAV turns towards the same side. In Figure 20b, the central AR-Drone (red) avoids the black AR-Drone (on left) and also goes to the right. Its current estimate of the other trapped AR-Drone is temporarily erratic beyond the anticipated bounds.
3. The second AR-Drone also becomes "trapped" along the border. As in Figure 20c, the two oscillate along the border until a collision occurs due to proximity.

This collision scenario does not occur with two MAVs because of the larger freedom of movement and the more accurate relative location estimate.

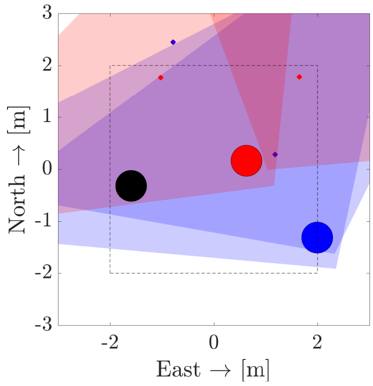
The failure mode described above may be tackled in different ways. One option is to increase mobility by increasing κ_α, prompting a lower asymptote for α_{CC}. The results in this article are based on a conservative choice ($\kappa_\alpha = 1$) so as to account for a worst case scenario, but this could be alleviated in order to resolve these situations. However, if not accompanied by an improvement in the relative localization estimate, this could increase the risk of collision. Alternatively, the linear relationship of the error with distance from Equation (14) can be changed into a piece-wise function in order to limit growth of the collision cone beyond a certain distance. This would allow for a higher error tolerance but *only* for objects beyond a certain estimated range. For example: $\kappa_\alpha = 1$ if $\rho_{ji} \leq 3$ and $\kappa_\alpha = 2$ if $\rho_{ji} > 3$. A third option would be to implement a selective obstacle avoidance method that prioritizes between obstacles.

C. Impact of Noise on System Performance

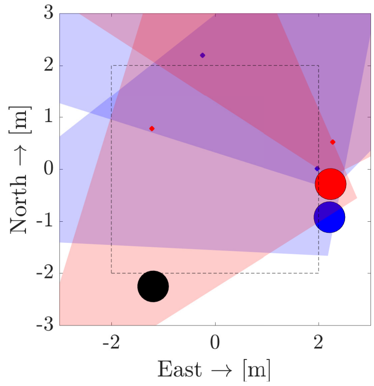
The real-world tests with two AR-Drones using off-board (Test #1) and on-board (Test #2) state estimation have shown how the performance of the relative localization algorithm is dependent on high-quality on-board estimates. The performance dropped from one collision in a cumulative $25min$ of flight to one every $\approx 3min$. To continue the discussion from the perspective of the RSSI measurements, this section investigates the extent to which an improvement in RSSI noise can lead to improved performance. In simulation, two case-studies are made. In the first case, the simulated RSSI noise is reduced from $5dB$ to $3dB$; lobes are still simulated. In the second case, RSSI noise is kept at $5dB$ but sensor lobes (all simulated disturbances) are removed. All EKF and collision avoidance parameters remain unchanged. The configurations analyzed are those with the lowest performance: 1, 2, 5, 6, 9,



(a) View #1. The blue AR-Drone (bottom right) moves towards the right. The red AR-Drone (middle) turns away from the black MAV and also towards the right.

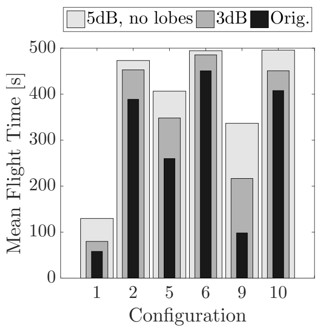


(b) View #2. The blue AR-Drone is trapped in the bottom right corner. The red AR-Drone continues towards the right.

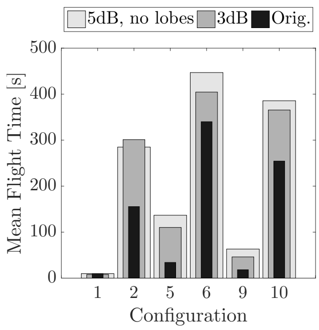


(c) View #3. The blue and red AR-Drones are both trapped at the right edge of the arena and begin alternately invoking M1 and M2 (see Section IV-A). This ends with a collision.

Fig. 20: Chronological depiction (left to right) of a collision case in a real-world flight with 3 AR-Drones. Large circles indicate the ground-truth position in the arena. The collision cones of the blue and the red MAVs are shown. The blue and red diamonds indicate the current estimates by the blue and the red MAV, respectively.



(a) Improvements in system performance with two MAVs when noise/disturbances are reduced.



(b) Improvements in system performance with three MAVs when noise/disturbances are reduced.

Fig. 21: Improvements in system performance against original results for lowest performing configurations (black, most narrow) when noise is reduced from $5dB$ to $3dB$ (dark gray, mid narrow) or when lobes are removed (dark gray, least narrow). The left figure is for a system with two MAVs and right figure is for a system with three MAVs.

10. The results are shown in Figure 21. It is systematically observed that removing the antenna lobes improves the performance. A lower noise also improves results, yet the impact is (generally) lower than antenna lobes. The lower error in relative position estimates successfully translates to a more successful collision avoidance system. This shows that performance could be improved further if operating in cleaner environments or if using higher quality sensors.

VII. CONCLUSION

We have shown the validity of Bluetooth as a relative localization sensor that can be used on-board of MAVs operating in a team. For MAVs that are already equipped with this technology, this enables swarm behavior *without* the need of a dedicated sensors. Intra-swarm collisions, a leading failure condition for MAVs flying in a limited space, can be successfully addressed by this technology provided that a collision avoidance system is used that properly encapsulates the errors involved.

In real world tests, two ARDrones 2.0 flying in in a $4m \times 4m$ space only collided once over a cumulative flight-time of $25min$. With three ARDrones 2.0, all else being equal, time between collisions was $\approx 3min$. The drop was due to increased disturbance and airspace density. When the AR-Drones were made to estimate their own velocity on-board using optical flow, two AR-Drones collided approximately every $3min$ as a result of the disturbances in the on-board velocity estimate. Simulation trials have shown that smaller MAVs in the same space would generally lead to lower collision rates.

The combined relative localization/collision avoidance system as presented and tested in this paper can be improved in different ways. Aside from hardware improvements, more investigations are advised in order to reliably reduce the error of the current relative localization filter. If this is done, it can translate into a higher freedom of movement for the MAVs without introducing higher risk in the system. Otherwise, the introduction of an additional strategy to deal with the avoidance of multiple team members should also improve performance when flying with 3 or more MAVs.

REFERENCES

[1] B. B. Mohr and D. L. Fitzpatrick, "Micro air vehicle navigation system," *Aerospace and Electronic Systems Magazine, IEEE*, vol. 23, no. 4, pp. 19–24, 2008.

[2] D. Scaramuzza, M. C. Achtelik, L. Doitsidis, F. Friedrich, E. Kosmatopoulos, A. Martinelli, M. W. Achtelik, M. Chli, S. Chatzichristofis, L. Kneip *et al.*, "Vision-controlled micro flying robots: from system design to autonomous navigation and mapping in gps-denied environments," *Robotics & Automation Magazine, IEEE*, vol. 21, no. 3, pp. 26–40, 2014.

[3] K. Alexis, G. Nikolakopoulos, A. Tzes, and L. Dritsas, "Coordination of helicopter uavs for aerial forest-fire surveillance," in *Applications of intelligent control to engineering systems*. Springer, 2009, pp. 169–193.

[4] M. Achtelik, M. Achtelik, Y. Brunet, M. Chli, S. Chatzichristofis, J.-D. Decotignie, K.-M. Doth, F. Fraundorfer, L. Kneip, D. Gurdan *et al.*, "Sfly: Swarm of micro flying robots," in *Intelligent Robots and Systems (IROS), 2012 IEEE/RSJ International Conference on*. IEEE, 2012, pp. 2649–2650.

[5] I. Sa and P. Corke, "Vertical infrastructure inspection using a quadcopter and shared autonomy control," in *Field and Service Robotics*. Springer, 2014, pp. 219–232.

[6] M. A. Kovacina, D. Palmer, G. Yang, and R. Vaidyanathan, "Multi-agent control algorithms for chemical cloud detection and mapping using unmanned air vehicles," in *Intelligent Robots and Systems, 2002. IEEE/RSJ International Conference on*, vol. 3. IEEE, 2002, pp. 2782–2788.

[7] B. Remes, P. Esden-Tempski, F. Van Tienen, E. Smeur, C. De Wagter, and G. De Croon, "Lisa-s 2.8 g autopilot for gps-based flight of mavs," in *IMAV 2014: International Micro Air Vehicle Conference and Competition 2014, Delft, The Netherlands, August 12-15, 2014*. Delft University of Technology, 2014.

[8] Y. Mulgaonkar, G. Cross, and V. Kumar, "Design of small, safe and robust quadrotor swarms," in *Robotics and Automation (ICRA), 2015 IEEE International Conference on*. IEEE, 2015, pp. 2208–2215.

[9] A. Kushleyev, D. Mellinger, C. Powers, and V. Kumar, "Towards a swarm of agile micro quadrotors," *Autonomous Robots*, vol. 35, no. 4, pp. 287–300, 2013.

[10] T. Szabo, "Autonomous collision avoidance for swarms of mavs based solely on rssi measurements," Master's thesis, Delft University of Technology, 2015.

[11] V. Crespi, A. Galstyan, and K. Lerman, "Top-down vs bottom-up methodologies in multi-agent system design," *Autonomous Robots*, vol. 24, no. 3, pp. 303–313, 2008.

[12] A. Iyer, L. Rayas, and A. Bennett, "Formation control for cooperative localization of mav swarms," in *Proceedings of the 2013 international conference on Autonomous agents and multi-agent systems*. International Foundation for Autonomous Agents and Multiagent Systems, 2013, pp. 1371–1372.

[13] M. Varga, "Fixed-wing drones for communication networks," Ph.D. dissertation, ÉCOLE POLYTECHNIQUE FÉDÉRALE DE LAUSANNE, 2016.

[14] G. Vásárhelyi, C. Virágh, G. Somorjai, N. Tarcai, T. Szörényi, T. Nepusz, and T. Vicsek, "Outdoor flocking and formation flight with autonomous aerial robots," in *2014 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2014, pp. 3866–3873.

[15] S. Shen, N. Michael, and V. Kumar, "Autonomous multi-floor indoor navigation with a computationally constrained mav," in *Robotics and automation (ICRA), 2011 IEEE international conference on*. IEEE, 2011, pp. 20–25.

[16] N. Michael, D. Mellinger, Q. Lindsey, and V. Kumar, "The grasp multiple micro-uav testbed," *Robotics & Automation Magazine, IEEE*, vol. 17, no. 3, pp. 56–65, 2010.

[17] A. Ledergerber, M. Hamer, and R. D'Andrea, "A robot self-localization system using one-way ultra-wideband communication," in *Intelligent Robots and Systems (IROS), 2015 IEEE/RSJ International Conference on*. IEEE, 2015, pp. 3131–3137.

[18] J. Faigl, T. Krajnik, J. Chudoba, L. Preucil, and M. Saska, "Low-cost embedded system for relative localization in robotic swarms," in *Robotics and Automation (ICRA), 2013 IEEE International Conference on*. IEEE, 2013, pp. 993–998.

[19] H. Ho, C. De Wagter, B. Remes, and G. de Croon, "Optical-flow based self-supervised learning of obstacle appearance applied to mav landing," *arXiv preprint arXiv:1509.01423*, 2015.

[20] P. Conroy, D. Bareiss, M. Beall, and J. van den Berg, "3-d reciprocal collision avoidance on physical quadrotor helicopters with on-board sensing for relative positioning," *arXiv preprint arXiv:1411.3794*, 2014.

[21] S. Roelofsen, D. Gillet, and A. Martinoli, "Reciprocal collision avoidance for quadrotors using on-board visual detection," in *Intelligent Robots and Systems (IROS), 2015 IEEE/RSJ International Conference on*. IEEE, 2015, pp. 4810–4817.

[22] T. Nageli, C. Conte, A. Domahidi, M. Morari, and O. Hilliges, "Environment-independent formation flight for micro aerial vehicles," in *Intelligent Robots and Systems (IROS 2014), 2014 IEEE/RSJ International Conference on*. IEEE, 2014, pp. 1141–1146.

[23] K. McGuire, G. de Croon, C. de Wagter, B. Remes, K. Tuyls, and H. Kappen, "Local histogram matching for efficient optical flow computation applied to velocity estimation on pocket drones," *arXiv preprint arXiv:1603.07644*, 2016.

[24] C. De Wagter, S. Tijmons, B. D. Remes, and G. C. de Croon, "Autonomous flight of a 20-gram flapping wing mav with a 4-gram onboard stereo vision system," in *Robotics and Automation (ICRA), 2014 IEEE International Conference on*. IEEE, 2014, pp. 4982–4987.

[25] J. F. Roberts, T. Stirling, J.-C. Zufferey, and D. Floreano, "3-d relative positioning sensor for indoor flying robots," *Autonomous Robots*, vol. 33, no. 1-2, pp. 5–20, 2012.

[26] M. Basiri, "Audio-based positioning and target localization for swarms of micro aerial vehicles," 2015.

[27] M. Basiri, F. Schill, D. Floreano, and P. U. Lima, "Audio-based localization for swarms of micro air vehicles," in *Robotics and Automation (ICRA), 2014 IEEE International Conference on*. IEEE, 2014, pp. 4729–4734.

[28] M. Basiri, F. Schill, P. Lima, and D. Floreano, "On-board relative bearing estimation for teams of drones using sound," *IEEE Robotics and Automation Letters*, vol. 1, no. 2, pp. 820–827, 2016.

[29] C. Lehnert and P. Corke, "μav-design and implementation of an open source micro quadrotor," *AC on Robotics and Automation, Eds*, 2013.

[30] K. Townsend, C. Cufí, R. Davidson *et al.*, *Getting started with Bluetooth low energy: Tools and techniques for low-power networking*. O'Reilly Media, Inc., 2014.

[31] W.-S. Soh *et al.*, "A comprehensive study of bluetooth signal parameters for localization," in *Personal, Indoor and Mobile Radio Communications, 2007. PIMRC 2007. IEEE 18th International Symposium on*. IEEE, 2007, pp. 1–5.

[32] P. Fiorini and Z. Shiller, "Motion planning in dynamic environments using velocity obstacles," *The International Journal of Robotics Research*, vol. 17, no. 7, pp. 760–772, 1998.

[33] J. Guzzi, A. Giusti, L. M. Gambardella, G. Di Caro *et al.*, "Local reactive robot navigation: A comparison between reciprocal velocity obstacle variants and human-like behavior," in *Intelligent Robots and Systems (IROS), 2013 IEEE/RSJ International Conference on*. IEEE, 2013, pp. 2622–2629.

[34] H. No, A. Cho, and C. Kee, "Attitude estimation method for small uav under accelerative environment," *GPS Solutions*, vol. 19, no. 3, pp. 343–355, 2015.

[35] M. H. Afzal, V. Renaudin, and G. Lachapelle, "Magnetic field based heading estimation for pedestrian navigation environments," in *Indoor Positioning and Indoor Navigation (IPIN), 2011 International Conference on*. IEEE, 2011, pp. 1–10.

[36] R. W. Beard, "State estimation for micro air vehicles," in *Innovations in Intelligent Machines-1*. Springer, 2007, pp. 173–199.

[37] A. M. Sabatini and V. Genovese, "A stochastic approach to noise modeling for barometric altimeters," *Sensors*, vol. 13, no. 11, pp. 15 692–15 707, 2013.

[38] K. Shilov, "The next generation design of autonomous mav flight control system smartap," in *IMAV 2014: International Micro Air Vehicle Conference and Competition 2014, Delft, The Netherlands, August 12-15, 2014*. Delft University of Technology, 2014.

[39] F. Kendoul, K. Nonami, I. Fantoni, and R. Lozano, "An adaptive vision-based autopilot for mini flying machines guidance, navigation and control," *Autonomous Robots*, vol. 27, no. 3, pp. 165–188, 2009.

[40] F. Kendoul, I. Fantoni, and K. Nonami, "Optic flow-based vision system for autonomous 3d localization and control of small aerial vehicles," *Robotics and Autonomous Systems*, vol. 57, no. 6, pp. 591–602, 2009.

[41] A. Martinelli and R. Siegwart, "Observability analysis for mobile robot localization," in *Intelligent Robots and Systems, 2005.(IROS 2005). 2005 IEEE/RSJ International Conference on*. IEEE, 2005, pp. 1471–1476.

[42] A. Martinelli, F. Pont, and R. Siegwart, "Multi-robot localization using relative observations," in *Robotics and Automation, 2005. ICRA 2005. Proceedings of the 2005 IEEE International Conference on*. IEEE, 2005, pp. 2797–2802.

[43] O. De Silva, G. K. Mann, R. G. Gosine *et al.*, "Relative localization with symmetry preserving observers," in *Electrical and Computer*

Engineering (CCECE), 2014 IEEE 27th Canadian Conference on. IEEE, 2014, pp. 1–6.
[44] A. Cornejo and R. Nagpal, "Distributed range-based relative localization of robot swarms," in *Algorithmic Foundations of Robotics XI*. Springer, 2015, pp. 91–107.
[45] J. S. Seybold, *Introduction to RF propagation*. John Wiley & Sons, 2005.
[46] A. Kushki, K. Plataniotis, and A. Venetsanopoulos, "Indoor positioning with wireless local area networks (wlan)," in *Encyclopedia of GIS*. Springer, 2008, pp. 566–571.
[47] T. S. Rappaport *et al.*, *Wireless communications: principles and practice*. prentice hall PTR New Jersey, 1996, vol. 2.
[48] J. Svečko, M. Malajner, and D. Gleich, "Distance estimation using rssi and particle filter," *ISA transactions*, vol. 55, pp. 275–285, 2015.
[49] Pulse, "W1049b datasheet version 1.1," 2008, accessed November 2015. [Online]. Available: www.cdiweb.com/datasheets/pulse/W1049B.pdf
[50] K. Nguyen and Z. Luo, "Evaluation of bluetooth properties for indoor localisation," in *Progress in Location-Based Services*. Springer, 2013, pp. 127–149.
[51] C. Caron, D. Chamberland-Tremblay, C. Lapierre, P. Hadaya, S. Roche, and M. Saada, "Indoor positioning," in *Encyclopedia of GIS*. Springer, 2008, pp. 553–559.
[52] V. Malyavej, W. Kumkeaw, and M. Aorpimai, "Indoor robot localization by rssi/imu sensor fusion," in *Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology (ECTI-CON), 2013 10th International Conference on*. IEEE, 2013, pp. 1–6.
[53] N. Point, "Inc.: Optitrack-optical motion tracking solutions," 2009.
[54] D. Wilkie, J. Van den Berg, and D. Manocha, "Generalized velocity obstacles," in *Intelligent Robots and Systems, 2009. IROS 2009. IEEE/RSJ International Conference on*. IEEE, 2009, pp. 5573–5578.
[55] C. Powers, D. Mellinger, A. Kushleyev, B. Kothmann, and V. Kumar, "Influence of aerodynamics and proximity effects in quadrotor flight," in *Experimental Robotics*. Springer, 2013, pp. 289–302.
[56] D. Claes, D. Hennes, K. Tuyls, and W. Meeussen, "Collision avoidance under bounded localization uncertainty," in *Intelligent Robots and Systems (IROS), 2012 IEEE/RSJ International Conference on*. IEEE, 2012, pp. 1192–1198.
[57] J. Van den Berg, M. Lin, and D. Manocha, "Reciprocal velocity obstacles for real-time multi-agent navigation," in *Robotics and Automation, 2008. ICRA 2008. IEEE International Conference on*. IEEE, 2008, pp. 1928–1935.
[58] J. Snape, J. van den Berg, S. J. Guy, and D. Manocha, "Independent navigation of multiple mobile robots with hybrid reciprocal velocity obstacles," in *Intelligent Robots and Systems, 2009. IROS 2009. IEEE/RSJ International Conference on*. IEEE, 2009, pp. 5917–5922.
[59] ——, "The hybrid reciprocal velocity obstacle," *Robotics, IEEE Transactions on*, vol. 27, no. 4, pp. 696–706, 2011.
[60] J. Van Den Berg, S. J. Guy, M. Lin, and D. Manocha, "Reciprocal n-body collision avoidance," in *Robotics research*. Springer, 2011, pp. 3–19.
[61] G. De Croon, E. De Weerdt, C. De Wagter, B. Remes, and R. Ruijsink, "The appearance variation cue for obstacle avoidance," *Robotics, IEEE Transactions on*, vol. 28, no. 2, pp. 529–534, 2012.
[62] M. Quigley, K. Conley, B. Gerkey, J. Faust, T. Foote, J. Leibs, R. Wheeler, and A. Y. Ng, "Ros: an open-source robot operating system," in *ICRA workshop on open source software*, vol. 3, no. 3.2, 2009, p. 5.
[63] N. Koenig and A. Howard, "Design and use paradigms for gazebo, an open-source multi-robot simulator," in *Intelligent Robots and Systems, 2004.(IROS 2004). Proceedings. 2004 IEEE/RSJ International Conference on*, vol. 3. IEEE, 2004, pp. 2149–2154.
[64] J. Meyer, A. Sendobry, S. Kohlbrecher, U. Klingauf, and O. Von Stryk, "Comprehensive simulation of quadrotor uavs using ros and gazebo," in *Simulation, Modeling, and Programming for Autonomous Robots*. Springer, 2012, pp. 400–411.
[65] M. Mueller and A. Drouin, "Paparazzithe free autopilot. build your own uav," in *24th Chaos Communication Congress, Berliner Congress Center, Dec*, 2007, pp. 27–30.
[66] P. Brisset and G. Hattenberger, "Multi-uav control with the paparazzi system," in *HUMOUS 2008, Conference on Humans Operating Unmanned Systems*, 2008.
[67] Parrot, "Ar drone 2.0," 2012. [Online]. Available: http://ardrone2.parrot.com
[68] BlueGiga, "Bled112 bluetooth smart dongle," 2016. [Online]. Available: https://www.bluegiga.com/en-US/products/bled112-bluetooth-smart-dongle
[69] T. Gaugel, J. Mittag, H. Hartenstein, S. Papanastasiou, and E. G. Strom, "In-depth analysis and evaluation of self-organizing tdma," in *Vehicular Networking Conference (VNC), 2013 IEEE*. IEEE, 2013, pp. 79–86.
[70] A. Kallapur, I. Petersen, and S. Anavatti, "A discrete-time robust extended kalman filter," in *American Control Conference, 2009. ACC'09.* IEEE, 2009, pp. 3819–3823.
[71] J. Sasiadek and Q. Wang, "Sensor fusion based on fuzzy kalman filtering for autonomous robot vehicle," in *Robotics and Automation, 1999. Proceedings. 1999 IEEE International Conference on*, vol. 4. IEEE, 1999, pp. 2970–2975.
[72] E. A. Wan and R. Van Der Merwe, "The unscented kalman filter for nonlinear estimation," in *Adaptive Systems for Signal Processing, Communications, and Control Symposium 2000. AS-SPCC. The IEEE 2000*. Ieee, 2000, pp. 153–158.
[73] S. J. Julier, "The scaled unscented transformation," in *American Control Conference, 2002. Proceedings of the 2002*, vol. 6. IEEE, 2002, pp. 4555–4559.
[74] A. Assa and F. Janabi-Sharifi, "A kalman filter-based framework for enhanced sensor fusion," *Sensors Journal, IEEE*, vol. 15, no. 6, pp. 3281–3292, 2015.
[75] L. DAlfonso, W. Lucia, P. Muraca, and P. Pugliese, "Mobile robot localization via ekf and ukf: A comparison based on real data," *Robotics and Autonomous Systems*, vol. 74, pp. 122–127, 2015.
[76] F. M. Dekking, *A Modern Introduction to Probability and Statistics: Understanding why and how*. Springer Science & Business Media, 2005.

Part II

Appendices

Appendix A

Literature Review on Quad-Rotor Dynamics

Quad-rotors are highly non-linear, under-actuated systems operating in 6 Degrees of Freedom (DOF) (Azzam & Wang, 2010). This section explores their dynamics and models that can be used in order to represent them. Appendix A-1 defines the general parameters, namely: frames of refences and intertia matrix. Appendix A-2 presents the assumptions used when modeling the motion of MAVs. Models found in literature are introduced in Appendix A-3 (linear) and in Appendix A-4 (non-linear). Appendix A-5 provides some concluding remarks.

A-1 Frames of reference

It is important to define two frames of reference to be used in this analysis: the body frame of reference $\mathcal{F}_B$, and the Earth-fixed (inertial) frame of reference $\mathcal{F}_E$. An illustration of the angular relationships between the two frames is presented in Figure A-1. Both systems abide to the North-East-Down (NED) convention. The rotation matrix from $\mathcal{F}_E$ to $\mathcal{F}_B$, denoted $\mathbf{R_{BE}}$, is given by Eq. A-1.

$$\mathbf{R_{BE}}(\phi,\theta,\psi) = \begin{bmatrix} c_\psi c_\theta & c_\psi s_\theta s_\phi - s_\psi c_\phi & c_\psi s_\theta c_\phi + s_\psi s_\phi \\ s_\psi c_\theta & s_\psi s_\theta s_\phi + c_\psi c_\phi & s_\psi s_\theta c_\phi - s_\phi c_\psi \\ -s_\theta & c_\theta s_\phi & c_\theta c_\phi \end{bmatrix} \tag{A-1}$$

Where ϕ is the roll angle, θ is the pitch angle, and ψ is the yaw angle. Note that, for compactness, $c_x = \cos x$ and $s_x = \sin x$. As Eq. A-1 is orthogonal (Phang, Cai, Chen, & Lee, 2012), the inverse transformation $\mathbf{R_{EB}}$ is given by its transpose, $\therefore \mathbf{R_{EB}} = \mathbf{R_{BE}^T}$.

The inertial matrix for a rotor is $J \in \mathbb{R}^3$, and is defined as Eq. A-2 with respect to $\mathcal{F}_B$. This matrix is to be simplified to a diagonal matrix thanks to an assumption elaborated in the next section.

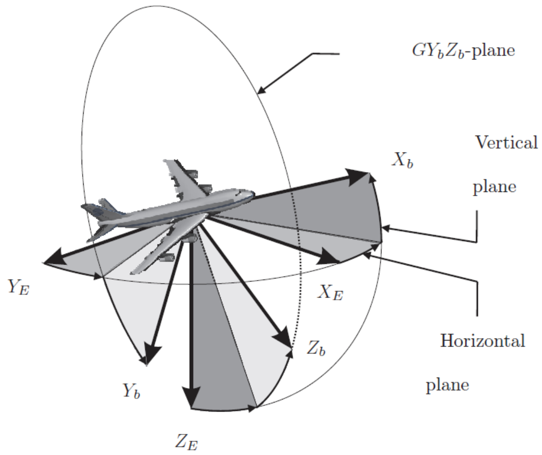


Figure A-1: Angular relationship between Body Frame of Reference $\mathcal{F}_B$ (denoted with lower case b in the figure) and Earth-fixed inertial frame $\mathcal{F}_E$ depicted on an aircraft (Mulder et al., 2013).

$$\mathbf{J} = \begin{bmatrix} J_{xx} & -J_{xy} & -J_{xz} \\ -J_{yz} & J_{yy} & -J_{yz} \\ -J_{zx} & -J_{zy} & J_{zz} \end{bmatrix} \tag{A-2}$$

A-2 General assumptions

The general assumptions in each model discussed in the following sections of this chapter are:

A1. **Constant mass.** The MAV features a constant mass throughout its full flight time. This assumption is valid as the MAVs are battery powered and do not rely on fuel.

A2. **Aligned thrust vector.** The thrust generated by the 4 on-board propellers are perfectly aligned with the z_B axis.

A3. **Rigidity.** Following from assumption A2, the MAV is assumed to be a fully rigid structure. Pounds, Mahony, and Corke (Pounds et al., 2006) has performed a modeling of blade flapping, included within the Newton-Euler model, by observing their first harmonic. The flapping introduces small lift force components into the x_B and y_B axis. However, this level of detail is not deemed necessary within the context of this research.

A4. **Symmetry.** The mass distribution and structural configuration of the quad-rotor MAV is perfectly symmetrical about the x_B-z_B and y_B-z_B planes. This gives $J_{xy} = J_{yx} = J_{xz} = J_{zx} = J_{yz} = J_{zy} = 0$ (K. U. Lee, Yun, Chang, Park, & Choi, 2011), and $J_{xx} = J_{yy}$.

A5. **Center of gravity at $\mathcal{F}_B$ origin**. The Center of Gravity (CG) coincides with the origin of $\mathcal{F}_B$ frame.

A-3 Linear models

Despite quad-rotors being inherently non-linear systems, it is still possible to represent their controlled system dynamics with linear systems to a certain degree of accuracy. This section presents two options for doing so.

A-3-1 Linearized aerodynamic model

Gremillion and Humbert (Gremillion & Humbert, 2010) have developed a linear quad-rotor model by determining a set of coefficients that attempt to linearize the dynamic interactions of the system. This system (Eq. A-3) is based on the model of a helicopter, but has been altered and adjusted to be suitable for the general case.

$$\begin{bmatrix}\dot{u}\\\dot{v}\\\dot{w}\\\dot{p}\\\dot{q}\\\dot{r}\\\dot{\phi}\\\dot{\theta}\\\dot{\psi}\end{bmatrix}=\underbrace{\begin{bmatrix}X_u&0&0&0&0&0&0&X_\theta&0\\0&Y_v&0&0&0&0&Y_\phi&0&0\\0&0&Z_w&0&0&0&0&0&0\\0&0&0&L_p&0&0&L_\phi&0&0\\0&0&0&0&M_q&0&0&M_\theta&0\\0&0&0&0&0&N_r&0&0&0\\0&0&0&\phi_p&0&0&0&0&0\\0&0&0&0&\theta_q&0&0&0&0\\0&0&0&0&0&\psi_r&0&0&0\end{bmatrix}}_{matrix\ A}\begin{bmatrix}u\\v\\w\\p\\q\\r\\\phi\\\theta\\\psi\end{bmatrix}+\underbrace{\begin{bmatrix}0&0&0&0\\0&0&0&0\\0&0&0&Z_r\\L_{lat}&0&0&0\\0&M_{lon}&0&0\\0&0&N_{yaw}&0\\\phi_{lat}&0&0&0\\0&\theta_{lon}&0&0\\0&0&\psi_{yaw}&0\end{bmatrix}}_{matrix\ B}\begin{bmatrix}\delta_{lat}\\\delta_{lon}\\\delta_{yaw}\\\delta_{thr}\end{bmatrix} \tag{A-3}$$

All variables in matrix A are aerodynamic derivatives that model the linearized dynamics of the model, which are to be experimentally estimated. All variables in matrix B are control derivatives, also to be estimated. u, v, w are the velocities expressed in $\mathcal{F}_B$. p, q, and r are the rotation rates around the $\mathcal{F}_B$ axis. ϕ, θ, and ψ are the roll, pitch, and yaw angles respectively. δ_{lat}, δ_{lon}, δ_{yaw}, and δ_{thr} are the input changes in the lateral position, longitudinal position, yaw, and thrust. Refer to the original source for a more in-depth account of all variables in the A and B matrices.

Advantages and disadvantages The use of this model, albeit advantaged by the linearity, falls short on a number of terms. One disadvantage of using this approach in practice is that it requires the estimation of a large number of control coefficients via system identification methods. Gremillion and Humbert performed this identification by using Vicon data coupled with NASA Langley's System Identification Programs for Aircraft (SIDPAC), which determines a least-squares solution from a pool of potential candidate regressors. Therefore, albeit the model is relatively simple, its practical usage requires thorough parameter estimation based on several flight data from different maneuvers. A further disadvantage is that this model is linearized about hover, and thus deemed more suitable for the linearized condition in a near-hover state.

A-3-2 Linearized kinematic model

A second option is to assume that all dynamics are controlled at lower levels, and thus to only model the high-level system kinematics. Szabo (Szabo, 2015) has done this by modeling the controlled quad-rotor as a second order linear system. The Micro Areal Vehicle (MAV) specific parameters (damping coefficient ζ and oscillation frequency ω_n) have been extracted experimentally, and the state equation of the model is as given in Eq. A-4, where: h is the height from the ground ($h = -z_E$), $p_E = \left[x_E, y_E, z_E\right]^T$ is the position of the vehicle in $\mathcal{F}_E$, t_c is the model time constant, Ψ_c is the heading of the quad-rotor, and h_c is the commanded height.

$$\begin{bmatrix} \dddot{x}_E \\ \ddot{x}_E \\ \dot{x}_E \\ \dddot{y}_E \\ \ddot{y}_E \\ \dot{y}_E \\ \dot{h} \end{bmatrix} = \begin{bmatrix} -2\zeta\omega & -\omega^2 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -2\zeta\omega & -\omega^2 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & -1/t_c \end{bmatrix} \begin{bmatrix} \ddot{x}_E \\ \dot{x}_E \\ x_E \\ \ddot{y}_E \\ \dot{y}_E \\ y_E \\ h \end{bmatrix} + \begin{bmatrix} \omega^2 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & \omega^2 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1/t_c \end{bmatrix} \begin{bmatrix} \left|\dot{\vec{p}}_E\right| \cdot \cos\left(\Psi_c\right) \\ \left|\dot{\vec{p}}_E\right| \cdot \cos\left(\Psi_c\right) \\ h_c \end{bmatrix} \tag{A-4}$$

Advantages and disadvantages This model is capable of re-creating the motion of a controlled element with the estimation of only two parameters. It is attractive due to its high simplicity, and can serve as an effective method for simulation of certain maneuvers. However, the model fails to account for the attitude of the unit, which can be a significant disadvantage, as well as coupling between the different axis.

A-4 Non-linear Newton-Euler model

A common quad-rotor model used in literature is based on the Newton-Euler model for 6 DOF motion, which describes the forces and torques experienced by the body as a function of the dynamics. The model is as presented in Eq. A-5 (Bouabdallah, Murrieri, & Siegwart, 2004) (Azzam & Wang, 2010), where: $\vec{F}_B$ is the force experienced by the rotor in $\mathcal{F}_B$, $\vec{\tau}_B$ is the torque experienced by the rotor in $\mathcal{F}_B$, $\vec{p_B}$ is the acceleration experienced by the rotor in $\mathcal{F}_B$, $\vec{\omega}$ is the angular velocity by the rotor in $\mathcal{F}_B$, $\mathbf{J}$ is the inertia matrix from Eq. A-2, $\mathbf{I_{3\times3}}$ is a 3×3 identity matrix.

$$\begin{bmatrix} \vec{F}_B \\ \vec{\tau}_B \end{bmatrix} = \begin{bmatrix} m\mathbf{I_{3\times3}} & 0 \\ 0 & \mathbf{J} \end{bmatrix} \begin{bmatrix} \ddot{\vec{p}}_B \\ \vec{\tau}_B \end{bmatrix} + \begin{bmatrix} \vec{\omega} \times m\dot{\vec{p}}_B \\ \vec{\omega} \times \mathbf{J}\vec{\omega} \end{bmatrix} \tag{A-5}$$

This format expresses the forces as a function of the dynamics in the body frame $\mathcal{F}_B$ for any body in 6 DOF. A quad-rotor, however, is an under-actuated body that can only produce a thrust force along the z_B axis (see assumption A2). It is then useful to apply the assumption and reverse the expressions such that the motion is expressed in an inertial frame $\mathcal{F}_E$ as function of the forces and torques produced by the four rotors, as in Eq. A-6 (D. Lee et al., 2013),

$$\begin{aligned} m\ddot{\vec{p}}_E &= -F_T \mathbf{R_{EB}} \vec{e}_3 + mg\vec{e}_3 \\ \mathbf{J}\dot{\vec{\omega}} + \mathbf{S}(\tilde{\omega})\mathbf{J}\vec{\omega} &= \vec{\tau}_B \\ \dot{\mathbf{R}}_{\mathbf{EB}} &= \mathbf{R_{EB}}\mathbf{S}(\tilde{\omega}), \end{aligned} \tag{A-6}$$

where: F_T is the combined thrust force by all motors (Eq. A-8), g is the gravitational acceleration, $e_3 = \begin{bmatrix}0, & 0, & 1\end{bmatrix}^T$, $S(\vec{\omega})$ is a skew matrix of $\vec{\omega}$. The force due to gravity acts along the z_E axis.

The impact on the z_B axis is given by

$$F_{g,B} = R_{BE} \begin{bmatrix} 0 \\ 0 \\ mg \end{bmatrix} = \begin{bmatrix} -mgs_\theta \\ mgc_\theta s_\phi \\ mgc_\theta c_\phi \end{bmatrix}. \tag{A-7}$$

The thrust generated by the four propellers, taking into account assumption A2, is equal to the the cumulative thrust from each, and is perfectly aligned along the z_B axis. The propellers are labeled *"front"* for the propeller along the positive x_B, *"right"* for the propeller along the positive y_B, *"back"* for the propeller along the negative x_B. *"left"* for the propeller along the negative y_B, Alternatively, they can be numbered 1 to 4, respectively.

$$F_T = \sum_{n=1}^{4} F_{T_i} = F_{T_{left}} + F_{T_{right}} + F_{T_{front}} + F_{T_{back}} \tag{A-8}$$

The torque $\vec{\tau}_B$ along the roll, pitch, and yaw axis, is then given by Eq. A-9, where: l is the arm length from the center of gravity to the propellers (assumed symmetric as in assumption A4), and f_t is a yaw drag factor. Note that for leveled flight $\tau_{right} = \tau_{left} = -\tau_{front} = -\tau_{back}$.

$$\vec{\tau}_B = \begin{bmatrix} \tau_\phi \\ \tau_\theta \\ \tau_\psi \end{bmatrix} = \begin{bmatrix} l \cdot \left(F_{T_{left}} - F_{T_{right}}\right) \\ l \cdot \left(F_{T_{front}} - F_{T_{back}}\right) \\ f_t \cdot (\tau_{right} + \tau_{left} + \tau_{front} + \tau_{back}) \end{bmatrix} \tag{A-9}$$

Advantages and disadvantages The key advantage of this model is its higher fidelity. Unlike the linear models expressed in Appendix A-3, this model focuses on describing the uncontrolled dynamics of the system based directly on its physical quantities. The downside of a model of this type is that it requires a non-linear numerical method to make predictions on future states, which can be computationally demanding. This is not predicted to be issue if the model is used during development for testing purposes or to employ off-line optimization schemes — in these scenarios the computation is not time-critical and can also be run on a state-of-the-art computers. However, if a non-linear simulation needs to be run on on-board (thus on a low-end chip) and in a time-critical environment, there may be complications. This is dependent on several factors including but not limited to: the discretization of time within the simulation, the desired fidelity/accuracy that needs to be reached, and the required update frequency of the control-scheme requiring the simulation. Furthermore, as the model does not take into account the control of the quad-rotor, an accurate simulation would then also require the inclusion of a model of the controller.

A-5 Conclusion and considerations for model implementation

This chapter presented three separate options for modeling the motion of a quad-rotor MAV. Each model exhibits a set of advantages and disadvantages. The general trend that is observed is that a more accurate model comes at the cost of increased implementation complexity as well as increased computational costs. The advantages of higher accuracy become more apparent for the analysis of short-term motion such as may be required during state-estimation (Leishman, Macdonald, Beard, & McLain, 2014) (see Appendix B-1). Lower fidelity models have been used to model the behavior of a system over longer time periods without requiring an in-depth analysis of lower-level system parameters (Szabo, 2015). Low fidelity algorithms provide a simple solution for the development and testing of algorithms for which it is not necessary to obtain data that is entirely representative of the ground-truth motion. For instance, the testing of a localization algorithms in simulation will not require high-fidelity motion parameters in order to establish its merits within the given context, provided that the localization algorithm is independent of the dynamics. During the analysis of obstacle avoidance motion, however, one may argue that there is a need for a higher fidelity model.

Appendix B

Literature Review on On-board Measurement of Own State

Prior to the analysis of methods for sensing and localization of other swarm members, it is useful to determine and discuss the capabilities of the on-board sensors that are used for self-state estimation. This is with the aim of determining what states the MAVs could reliably share with each-other during a localization task. MAVs feature 12 states (Beard, 2007) (Bry, Bachrach, & Roy, 2012), namely:

- x_E, y_E, z_E: the positions with respect to a fixed inertial frame of reference (f.o.r.) $\mathcal{F}_E$;
- u, v, w: the velocities with respect to a North-East-Down body f.o.r. $\mathcal{F}_B$;
- ϕ, θ, ψ: the Euler angles defining the axis rotations between $\mathcal{F}_E$ and $\mathcal{F}_B$;
- p, q, r: the rotations along x_B, y_B, and z_B, respectively.

These 12 states need to be measured and/or estimated using on-board sensors and processing. The relevant sensors expected to be on-board of the envisioned platform, as extracted from (Remes et al., 2014), are: an Inertial Measurement Unit (IMU), a magnetometer, a pressure sensor, Global Positioning System (GPS), and a mono/stereo camera.

The analysis that follows focuses on the capabilities and raw data that can be provided by each sensor, as in Appendix B-1-Appendix B-5, and the current state of the art. Appendix B-6 then uses the information to conclude with a summary of the relevant sensors that are considered in this research, their utility, and their applications. It should be noted that the scope of the thesis work is beyond the implementation of the methods described in this section. Nevertheless, a study into the state-of-the-art is useful in order to aid with determining the feasibility of the methods for relative localization and collision avoidance.

B-1 Inertial Measurement Unit (IMU)

The IMU as treated in this report consists of two sub-units: a *3-axis accelerometer* and a *3-axis gyroscope*.

3-axis accelerometer The accelerometer measures the accelerations in the body frame of reference $\mathcal{F}_B$.

$$\begin{bmatrix} \ddot{x_B} \\ \ddot{y_B} \\ \ddot{z_B} \end{bmatrix}_{measured} = R_{BE} \left(\begin{bmatrix} \ddot{x_E} \\ \ddot{y_E} \\ \ddot{z_E} \end{bmatrix} - g \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \right) + \vec{\beta}_a + \vec{\eta}_a \tag{B-1}$$

In Eq. B-1: $\vec{\beta}_a$ indicates the bias and $\vec{\eta}_a \in \mathcal{N}(0, \vec{\sigma}_a)$ indicates zero-mean Gaussian noise with standard deviation $\vec{\sigma}_a$. The accelerometer also includes the effects of gravitational acceleration g within its measurement.

3-axis gyroscope The gyroscope measures the rotations along the axis of the body-frame $\mathcal{F}_B$, thus providing direct measurements for p, q, and r.

$$\begin{bmatrix} p \\ q \\ r \end{bmatrix}_{measured} = \begin{bmatrix} p \\ q \\ r \end{bmatrix} + \vec{\beta}_g + \vec{\eta}_g \tag{B-2}$$

In Eq. B-2: $\vec{\beta}_g$ is the bias, and $\vec{\eta}_g \in \mathcal{N}(0, \vec{\sigma}_g)$ is a zero-mean Gaussian noise.

Extraction of position, velocity, and attitude Both the accelerometer and the gyroscope can be assumed to have a constant (or slow varying) bias term, and the noise is zero-mean Guassian, such that $\eta_a \in \mathcal{N}(0, \sigma_a)$ and $\eta_g \in \mathcal{N}(0, \sigma_g)$ (Beard, 2007) (Mahony, Kumar, & Corke, 2012). Bias increase becomes significant when the outputs are integrated due to the unpredictable accumulation errors in order to infer velocity, position, or attitude. This can be corrected for by fusing IMU measurements with measurements from other sensors (such as those from a magnetometer, GPS, or vision). However, a notable achievement for attitude and velocity estimation using *only* IMU measurements is found in the work by Leishman et al. (Leishman et al., 2014), building upon the same author's work from (Macdonald, Leishman, Beard, & McLain, 2014). The research concluded that accurate attitude and velocity estimates in the x_B and y_B axis can be performed using IMU-only measurements if the raw data is Kalman filtered using an accurate non-linear model. The model is similar to the Newton-Euler model from Eq. A-5 to A-6, but with the added inclusion of a term modeling the drag effects on planar velocity (otherwise generally considered negligible over other forces). Both papers condemn the lack of physically accurate IMU based attitude and velocity estimates on the fact that previous works have abstained from including drag forces within their model. Albeit this choice is generally justified by its negligible size for multi-rotor MAVs, their research shows notable improvements in the velocity and attitude estimates. This approach is also shared by Abeywardena, Kodagoda, Dissanayake, and Munasinghe in (Abeywardena et al., 2013).

B-2 Magnetometer

A magnetometer reads the total Earth magnetic field B in the body frame of reference $\mathcal{F}_B$, as in Eq. B-3, where: $\vec{B}_B$ and $\vec{B}_E$ are the magnetic fields in the frames $\mathcal{F}_B$ and $\mathcal{F}_E$ respectively,

R_{BE} is the relevant rotation matrix from $\mathcal{F}_E$ to $\mathcal{F}_B$ (see Eq. A-1), $\vec{\beta}_m$ is the disturbance vector, and $\vec{\eta}_m \in \mathcal{N}(0, \sigma_m)$ is the noise vector.

$$\vec{B}_B = R_{BE}\vec{B}_E + \vec{\beta}_m + \vec{\eta}_m \tag{B-3}$$

The yaw angle with respect to magnetic North can then be estimated with Eq. B-6, which is corrected for roll ϕ and pitch θ using Eq. B-4 (No, Cho, & Kee, 2015).

$$\vec{B}_E = R_{EB}(\phi, \theta, 0)\vec{B}_B \tag{B-4}$$

$$\psi = \arctan\left(\frac{B_{E_y}}{B_{E_x}}\right) + D \tag{B-5}$$

$$\tag{B-6}$$

B_x, B_y, and B_z are illustrated in Figure B-1. D is the declination angle of the Earth's magnetic field (this is a fixed value depending on the geographic location, but can be neglected if the yaw angle is used merely for comparative purposes).

Magnetometers usually feature low noise but are subject to magnetic disturbances. Particularly when used indoors, due to the magnetic anomalies that are bound to be present in the environment, they can often be subject to low accuracy at higher-frequencies as seen in (Sharp & Yu, 2014). The effect of such anomalies is depicted in Figure B-1 (Afzal, Renaudin, & Lachapelle, 2011). Magnetometer behavior opposes the general trend that is observed in gyroscopes, which instead feature *"good short-term accuracy, but due to instrumentation offsets, the integrated output will become progressively inaccurate over time"* (Sharp & Yu, 2014). The implication is that magnetometers provide good information at low frequencies, whereas data from the gyroscopes is useful in the higher frequencies. Therefore, one solution to the issue is to adopt a complementary filter as outlined by Pascoal, Kaminer, and Oliveira (Pascoal et al., 2000), of which the state-space formulation is given in Eq. B-7. This filter blends the yaw-rate measurements from a rate gyro (r_m) with the yaw measured from a magnetometer compass (ψ_m) to output an estimate $\hat{\psi}$. k_1 and k_2 are the filter gains, and r_d represents the gyroscope bias in r that is rejected by the system.

$$\begin{aligned} \begin{bmatrix} \dot{\hat{\psi}} \\ \dot{r_d} \end{bmatrix} &= \begin{bmatrix} -k_1 & 1 \\ -k_2 & 0 \end{bmatrix} \begin{bmatrix} \hat{\psi} \\ r_d \end{bmatrix} + \begin{bmatrix} k_1 \\ k_2 \end{bmatrix} \psi_m + \begin{bmatrix} 1 \\ 0 \end{bmatrix} r_m \\ \hat{\psi} &= \begin{bmatrix} 1 & 0 \end{bmatrix} \begin{bmatrix} \hat{\psi} \\ r_d \end{bmatrix} \end{aligned} \tag{B-7}$$

B-3 Pressure sensors

The pressure sensor measures the static pressure force P, modeled by Eq. B-8, where ρ is the density of the air and h is the altitude with respect to the ground (Beard, 2007). β_p is the bias and η_p is a zero-mean Gaussian noise, i.e. $\eta_p \in \mathcal{N}(0, \sigma_p)$.

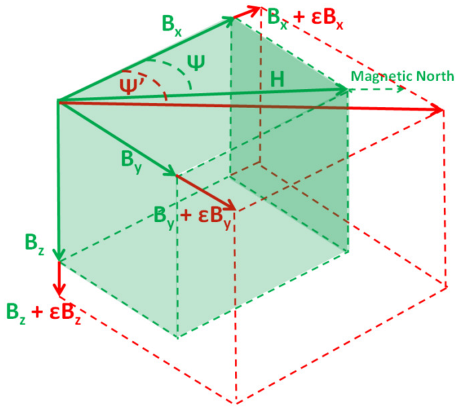


Figure B-1: Depiction of magnetometer error when estimating a magnetic field due to anomalies in the environment (Afzal et al., 2011). Vectors B_x, B_y, and B_z represent the magnetic field strength in a NED body frame $\mathcal{F}_B$, perturbed respectively by errors εB_x, εB_y, and εB_z. The result is a perturbation of the output to ψ' as opposed to the true value ψ.

$$P(h)_{measured} = \rho g h + \beta_p + \eta_p \tag{B-8}$$

β_p is constant and can be neglected if the sensor is calibrated with respect to a point $z_{E,C}$ (e.g. the ground), leading to:

$$P(\Delta z_E) = P(h)_{measured} - P(z_{E_0}) = \rho g \Delta z_E + \eta_p. \tag{B-9}$$

Pressure sensors on quad-rotor MAVs can return significant noise at both high and at low frequencies due to wind-gusts, potential air-pockets, and disturbance from the airflow during flight. A common technique to solving the noise-issue is by performing averaging of the signal in real-time (moving-average) (Sabatini & Genovese, 2013) (Shilov, 2014). In (Shilov, 2014), featuring a Bosch BMP085 barometer on a small MAV, the height measurement reached an accuracy of $\pm 80cm$.

B-4 Global Positioning System (GPS)

When available (i.e. outdoors), GPS is an invaluable resource for providing absolute positioning estimates. Furthermore, GPS is capable of providing velocity estimates (via differentiating the position observations) as well as heading information. These can be useful as raw measurements as well as to help estimate other states, such as in (Gebre-Egziabher & Elkaim, 2008), where Gebre-Egziabher and Elkaim develop an attitude determination algorithm by relying on GPS to provide velocity estimates. However, this thesis aims to operate in an indoor environment where GPS is unavailable. This sensor will hence not be considered further.

B-5 Vision

A monocular (single) or stereo (dual) camera can be used for several purposes including: navigation (Scaramuzza et al., 2014) (Kendoul, Nonami, Fantoni, & Lozano, 2009), state-estimation (De Croon, De Wagter, Remes, & Ruijsink, 2012) (Kendoul, Nonami, et al., 2009), and obstacle recognition (De Croon et al., 2013) (Corke, 2011). The constraints of this sensor are both hardware-dependent (e.g. image size in pixels, image sharpness, frame-rate), and software-dependent based on the characteristic of the vision processing algorithm. A digital image can be represented as a grid of $W \times H = N$ pixels. Where a point in the image is given by a co-ordinate u_{im}, v_{im}, and by convention $(u_{im}, v_{im}) = (0, 0)$ is the top left corner [1]. The image can be stored in gray-scale or color, albeit for the purposes of edge and point feature detection color is not found to greatly increase the performance despite the high impact on memory and processing load. A strong reliance on color can even be problematic if there is a lighting change between pictures (Corke, 2011).

B-5-1 Feature detection and matching

Two basic primary features can be extracted from an image: line features and point features. Line features are determined from peaks in an *edge image*. An edge image is a filtered image where only edges (areas of significant shading change which can be detected by a dedicated *kernel* [2]) are visible. Lines can be represented by the pair (ρ_{im}, θ_{im}) with Eq. B-10, where: θ_{im} is the clockwise angle with the horizontal, and ρ_{im} is the closest distance from $(u_{im}, v_{im}) = (0, 0)$ to the line (i.e. the length of the normal from the line to the diagonal of the picture, assuming the picture is squared).

$$v_{im} = -u_{im} \tan(\theta_{im}) + \frac{\rho_{im}}{\cos(\theta_{im})} \tag{B-10}$$

Point features are recognizable by their high image gradient in multiple direction (indicating intersecting lines) around a window. A point feature, or corner, needs to be sufficiently different from nearby pixels in order to be consistently recognizable. Table B-1 provides a few common corner strength detectors.

Point features may not fare well with scaling; the same feature will appear more prominent and detailed as it gets closer and thus may not be recognized as being the same. Two popular methods for feature recognition that are invariant to scaling, orientation, and lighting are Scale-Invariant Feature Transform (SIFT) by Lowe (Lowe, 1999) and Speeded Up Robust Feature (SURF) by Bay, Ess, Tuytelaars, and Van Gool (Bay et al., 2008). In both instances, features are detected as the maxima that emerges over several Guassian kernel convolutions on an image. SIFT looks at the differences within the sequence, whereas SURF looks at the Hessian of the Gaussian within the sequence. The scale of the feature is based on the cumulative standard deviation within the Guassian sequence where a maxima is still detectable. The feature orientation is taken by extracting the dominant direction of the gradient.

[1] The rest of this report shall make use of this convention when relevant.

[2] A kernel is a matrix that is convolved with the original picture in order to alter it. A popular edge detection kernel matrix is the Laplacian of Gaussian (LoG), also known as "mexican hat" due to the shape of its function if plotted in 3D (Corke, 2011) (Lowe, 1987).

B-5-2 Visual Odometry

Visual odometry refers to the use of vision to infer the odometry of a moving vehicle. This is based on the extracted camera rotation and translation between two different camera views of a set of matching features (underlining the need for good feature description).

Assume a point feature $\vec{p}_{im_1} = (u_{im_1}, v_{im_1}, 1)$ (the third member is due to the use of homogeneous coordinates) in one figure and $\vec{p}_{im_2} = (u_{im_2}, v_{im_2}, 1)$ in another. The relationship between the two is given by the *epipolar constraint* (Eq. B-11), where $\mathbf{F_{21}}$ is the *fundamental matrix* relating the two points. $\mathbf{F_{21}}$ is a function of the camera parameter matrix $\mathbf{K_{cam}}$, the rotation from camera 2 to camera 1 $\mathbf{R_{21}}$, and the translation from camera 2 to camera 1 $\vec{t_{21}}$.

$$\begin{aligned} \vec{p}_{im_2}^{T}\mathbf{F_{21}}\vec{p}_{im_1} &= 0 \\ \mathbf{F_{21}} &= f(\mathbf{K_{cam}}, \mathbf{R_{21}}, \vec{t_{21}}) \end{aligned} \tag{B-11}$$

An alternative epipolar constraint is expressed using the *essential matrix* $\mathbf{E_{21}}$ as in Eq. B-12. $\mathbf{E_{21}}$ does not have to be related to the camera parameters because $\vec{x}_{im}$ is equal to the point feature is $\vec{p}_{im}$ but already adjusted to the focal length of the camera(s).

$$\begin{aligned} \vec{x}_{im_2}^{T}\mathbf{E}_{21}\vec{x}_{im_1} &= 0 \\ \mathbf{E_{21}} &= f(\mathbf{R_{21}}, \vec{t_{21}}) \end{aligned} \tag{B-12}$$

A popular method for model generation via feature pair matching is RANdom SAmpling and Consensus (RANSAC). This method uses a set of randomly selected matching point features to extract the relative pose of one camera view with respect to the other, and then follows to test the model on all other matching pairs in order to establish its validity. The process is iterated multiple times and the model with the highest overall mark is taken. The fundamental matrix has 8 DOF and thus needs 8 matching pairs to produce a model (Corke, 2011) for the camera rotation/translation between two figures (note that RANSAC needs more feature pairs in order to then test the model's accuracy). The essential matrix, which is independent of camera based parameters, features 5 DOF (Corke, 2011) (Troiani, Martinelli, Laugier, & Scaramuzza, 2013), and thus only needs 5 matching feature pairs. Under the assumption of planar flight and with the aid of a gyroscope to provide the angular rates of motion, Troiani et al. (Troiani et al., 2013) have managed to determine that 1 point is sufficient for motion tracking using the essential matrix. Martinelli (Martinelli, 2011) provides a more general solution for a body moving in 6DOF, but then needing to track a feature over 4 intervals.

If assuming that features are located on a plane, then it is possible to estimate its relative model, described by $\mathbf{H}$, using *planar homography* as in Eq. B-13.

$$\vec{p}_{im_2} = \mathbf{H}\vec{p}_{im_1} \tag{B-13}$$

Once again, RANSAC can be used to extract and test a model over all the features. Note that only the predominant plane is found. Less dominant planes can be found subsequently by removing the points that are already attributed to another plane.

B-5-3 Attitude estimation

It is possible to estimate the attitude with respect to a plane by analyzing the features assumed to be on a plane using planar homography (Corke, 2011) or by feature tracking with IMU fusion (Martinelli, 2011). If flying outdoors, pitch and/or roll can also be estimated based on the horizon line. De Croon, De Wagter, et al. (De Croon, De Wagter, et al., 2012) do this by using a supervised learning algorithm (linear perceptrons) to establish the boundary between sky and non-sky pixels, showing how the process can be further reduced by the use of passive sub-sampling without significant losses. For *absolute* state estimation using *only* a visual sensor, the algorithm needs certain knowledge on the expected properties of the environment to look for (such as the existence and purpose of a horizon line, or the existance and purpose of a floor/wall). In indoor environments, these properties are in nature more unpredictable and difficult to define.

B-5-4 Optical Flow

An alternative to visual odometry is Optical Flow (OF). OF, unlike visual odometry, is not concerned with the tracking of relative position or velocity as absolute real-world quantities, but only with the relative *flow* of features along an image. Therefore, unlike the previously described works, there is no reliance on the programmed knowledge of features of the environment (such as object shapes, horizon lines, etc.). The core of the algorithm is bio-inspired from flying insects (Kendoul, Nonami, et al., 2009) and has been proven sufficient for navigation and more complex operations such as slope estimation in landing (De Croon et al., 2013) or obstacle detection (De Croon, De Weerdt, De Wagter, Remes, & Ruijsink, 2012) (the latter is explored in more detail in Appendix H). By fusing OF measurements with IMU measurements, Kendoul, Nonami, et al. (Kendoul, Nonami, et al., 2009) (Kendoul, Fantoni, & Nonami, 2009) have shown how OF can also be an effective tool for navigation. Both implementations feature bottom-facing cameras that extract motion data, which is then filtered with rotation measurements (using the gyroscope and accelerometer input from the IMU) in order to extract an estimate of motion with respect to certain tracked targets. When filtering the measurements together, the acceleration of the IMU can be related to the acceleration of the tracked targets in order to extract height from the ground. This then makes it possible to extract real-world velocity and (relative) position. Note that camera parameters (i.e. focal length) also need to be known. An implementation difference between the methods is that (Kendoul, Nonami, et al., 2009) estimates height above ground using a Recursive Least Squares (RLS) method, whereas (Kendoul, Fantoni, & Nonami, 2009) filters the data through an Extended Kalman Filter (EKF). Another notable difference between the two implementations is that (Kendoul, Nonami, et al., 2009) uses feature tracking within a small window of features starting from the center of the image, whereas (Kendoul, Fantoni, & Nonami, 2009) tracks optical flow using a block-matching algorithm. The implementation reach reliable results with maximum errors of $\approx 1m$ for altitude hold mode. The results in either case are comparable to GPS for horizontal navigation and *more accurate* than GPS for vertical navigation.

B-6 Conclusion on obtainable odometry data

It has been determined that the on-board sensors can be capable of providing the states described in the chapter introduction. The conclusions drawn from this chapter, with an eye towards data sharing between MAVs for the purpose of localization and collision avoidance, are listed below.

- An estimate for the velocity in the body frame can be achieved by using the on-board IMU (with potential fusion of other measurements). Using IMU only reading (Leishman et al., 2014) have reached Root Mean Squared Error (RMSE) of 0.67 m/s. This can be further improved by inclusion of more sensors within a filter, and by combining an IMU with the vision.
- Roll and pitch attitude estimation is possible using IMU, vision, or other sensors. This information is also considered necessary for the sake of stable flight. Leishman et al. have reached RMSE of 2.23° for pitch and roll (Leishman et al., 2014) using only an IMU. Additionally, vision can help avoid potential issues related to drift.
- Height from the ground can be directly measured with the pressure sensor. However, as each sensor is subject to a different bias, calibration with respect *to the same point for all MAVs* needs to be performed if height data is to be shared between different members of a swarm. This sensor is less precise, providing errors between $0.8m$ up to $1m$.
- Magnetometers can provide a measurement of heading that can be common to multiple MAVs. Issues arise due to disturbances, as two magnetometers may experience different disturbance due to the environment. Even if disturbances are not filtered out correctly as in Eq. B-7, it is possible to make the assumption that if the magnetometers are sufficiently close to each-other, then they should be subject to similar disturbances and thus their measurements should be comparable. Via complementing a gyroscope with a magnetometer, albeit not for MAV flight, Sharp and Yu reached a zero-mean error with a standard deviation of $\approx 2°$.

Table B-1: Summary of three corner strength detectors (Corke, 2011). $\mathbf{A}$ is the structure tensor matrix of a Guassian weight gradient map of the image. λ_1 and λ_2 are the eigenvalues of $\mathbf{A}$. $tr(A) = \lambda_1 + \lambda_2$. $\det A = \lambda_1 \lambda_2$. k is an arbitrary tuning factor.

Shi-Tomasi detector	$C_{ST}(u_{im}, v_{im}) = \min(\lambda_1, \lambda_2)$
Harris detector	$C_H(u_{im}, v_{im}) = \det \mathbf{A} - ktr(\mathbf{A})^2$
Noble detector	$C_N(u_{im}, v_{im}) = \frac{\det \mathbf{A}}{tr(\mathbf{A})}$

Appendix C

Literature Review on Signal-based Methods for Relative Range/Bearing Measurements

The first category of methods to obtain raw measurements of either relative range or relative bearing between MAVs is that of *signal-based* methods. This encloses all methods that rely on the transmission of a signal of any kind between the MAVs. These methods fall into the three types below.

- *Time-based*, which are extract the range between two antennas from a time measurement between two events (see Appendix C-1);
- *Angle of Arrival (AOA)*, in which a receiver must infer the bearing of a transmitter (see Appendix C-2);
- *Received Signal Strength (RSS)*, in which the signal strength (which diminishes over distance) is used to infer the range between a transmitter and a receiver (see Appendix C-3).

C-1 Range and localization from time-based methods

The basic time-based method is Time of Arrival (TOA), which measures the time it takes for a signal to travel from a transmitter to a receiver, from which the receiver can extract the range (Vossiek, Wiebking, Gulden, Weighardt, & Hoffmann, 2003; Youssef, 2008). At short distances, state-of-the-art methods rely on sound coupled with radio signals. The radio signal acts as a trigger and is received almost instantaneously whereas the audio signal, traveling at a lower speed, arrives with a certain delay (Spears et al., 2007) which is proportional to the distance.

TOA has also been extended to the following methods in order to circumvent the need to send an initial trigger signal:

- *Round-trip Time of Arrival (RTOA).* In this method, a device measures the time it takes for a signal to travel to a receiver and back in order to establish the range to it (Vossiek et al., 2003). This eliminates the need to send an initial radio signal. For accuracy at short distance, it is crucial that the initiating device also takes proper account of the processing time of the other device.
- *Time Difference of Arrival (TDOA).* A transmitter emits a signal, which is then received by at least three antennas in known locations (Youssef, 2008; Fischer, Dietrich, & Winkler, 2004). The receiving antennas use the difference between their different arrival times, from which trilateration can be performed to extract the location of the transmitter.

Notable uses of time-based methods in robotics are found in (Spears et al., 2007) and (Perkins et al., 2011), both using a speaker with a round reflector to allow the sound to propagate *omni-directionally.* It is crucial that the signal travels as omni-directionally as possible such that range can be measured accurately from all relative bearings (Spears et al., 2007). For MAVs, however, this is difficult to achieve due to the need for a reflector that reflects the signal omni-directionally in a 3D environment, and this inclusion may lead to a significant mass and size penalty.

It should be noted that when used in small indoor environments, signals may be reflected from any surfaces such as the ground, the walls, or the ceiling. When this happens, the receiver will receive multiple instances of the same signal at different times, causing an ambiguity. Other sounds in the environment may also be incorrectly interpreted (Perkins et al., 2011) (other sources may also be signals from other MAVs).

Due to the basic practical issues regarding receiver/transmitter size and mass (which would have to be added to the Lisa-S Ladybird MAV), and due to the implementation issues regarding confounding factors in indoor environments, time-based methods are not considered further in this work.

C-2 Bearing detection from a signal's angle of arrival

AOA is any method that relies on attempting to directly infer the relative bearing of a transmitter with respect to a receiver. AOA requires directional antennas (Vossiek et al., 2003), or a dedicated receiver that is capable of triangulating the position of the source (Basiri, Schill, Floreano, & Lima, 2014) (Xu, Ma, & Law, 2015).

State-of-the-art methods for robots and MAVs rely on the use of an array of microphones. The time delay between their individual reception of a sound signal allows for the estimation of the bearing of the source. This system has been already implemented on ground robots (Spears et al., 2007) and MAVs (Basiri, 2015; Basiri et al., 2014), achieving interesting results.

Audio-based microphone arrays like the one used by (Basiri et al., 2014) are sufficiently light to fall within the design driver of size and weight minimization. However, their use has only been proven at long range distances and in outdoors environments. Indoor performance has only been tested with a static observer tracking engine sounds from speakers in a room, but one may expect that an indoor environment may cause reflections that act as confounding signals. Furthermore, a significant deterrent for localization accuracy is the noise emitted from the propellers of the MAV itself. This has also been admitted by (Basiri et al., 2014), who was forced to temporarily turn-off engines during flight-tests in order to improve the quality of an estimate. This may be a possibility for fixed-winged MAVs but it is not for a quad-rotor.

These expected complications, combined with the need to include a microphone array and a speaker to a system which would otherwise benefit from any possible mass-reduction, are significant deterrents for angle-of-arrival based localization in indoor environments.

Speaker/microphone combinations are not the only option found in literature for AOA localization. Roberts, Stirling, Zufferey, and Floreano use a laser-scanner ball with several emitters and receivers in order to determine the angle of the oncoming signal (Roberts et al., 2012). Their custom-made sensor has a mass of 245.2 g, a diameter of 22 cm, and is estimated to consume 10 W of power, which is needed to ensure a sufficient number of IR sensors to span all horizontal and vertical bearings with a maximum error of 4.3°. This mass is more than 5 times higher than the whole Lisa-S Ladybird drone. A weight reduction in the order of magnitude of 100 times would be needed before a similar sensor could be considered acceptable.

C-3 Received Signal Strength

The power of electromagnetic waves decreases over distance, and it is then possible to calculate the distance between a receiver and a transmitter based on the signal strength difference [1]. This can be done when two antennas are communicating using signals such as Wi-Fi or Bluetooth, from which an indication of RSS can be extracted over a communication. Like other signals, the measured strength will also be affected by reflection in the environment, but as (unlike methods above, including AOA) it does not depend on time, the reflection cannot cause a confound; it is interpreted as noise.

Communication between the MAVs is allowed according to the constraints of this project. This is not an unlikely scenario in a swarm that must co-operate. It means that a signal, not intended for localization, may already be traveling between the MAVs. Therefore, unlike all other previous methods discussed, RSS does not require an additional sensor to be placed on-board (Lisa-S Ladybird drones are already equipped with a Bluetooth module for ground communication).

For localization tasks within a closed environment, there are two general methods to extract distance from RSS: *finger-printing* and *model-based*. Finger-printing relies on a pre-established map (or *finger-print*) of the RSS distribution in an environment in order to infer a location (provided multiple beacons are present to perform trilateration). Finger-printing circumvents the need to model the effect of potential disturbances on the signal propagation by merely including such disturbances in a map, assumed static. This approach is usually used for indoor localization using static beacons (Dahlgren & Mahmood, 2014), but it has also been used for swarm MAV flight. Benjamin, Erinc, and Carpin (Benjamin et al., 2015), for instance, used GPS enabled drones to construct a Wi-Fi RSS map that can then be used by several other drones that are not GPS enabled to fly around an inhabited area.

[1] A transmitter, in this context, may also be referred to as Access Point (AP).

The requirements of this thesis explicitly deny the possibility of placing beacons in an environment a-priori, as this would not allow the system to operate in an unknown environment. This means that the beacons *must be* the MAVs themselves, but these are non-static and render finger-printing impractical and non-versatile. The viable alternative to finger-printing is to rely on a model of the signal propagation around a source in order to estimate the distance of a transmitter based on the received RSS, this is known as a *model-based* method.

Appendix D

Literature Review on Relative Localization Methods

D-1 Relative Localization Framework

A relative pose measurement is one that provides raw data related to one or more of the following: relative position between robots, and relative attitude between robots. The definition of the physical quantities to be measured requires a formal outlining of the relative localization framework. This is based on the general ego-centric framework inspired by Howard, Matarić, and Sukhatme (Howard et al., 2003), and it formalizes the parameters needed to fully describe the pose between two robots.

Consider two MAVs $\mathcal{R}_i$ and $\mathcal{R}_j$ with body frames $\mathcal{F}_{B_i}$ and $\mathcal{F}_{B_j}$, respectively. Under this framework, the full relative pose of $\mathcal{R}_j$ with respect to $\mathcal{R}_i$ can be defined as

$$\vec{\mathcal{P}}_{ji} = \begin{bmatrix} x_{ji} & y_{ji} & z_{ji} & \phi_{ji} & \theta_{ji} & \psi_{ji} \end{bmatrix},$$

where x_{ji}, y_{ji}, and z_{ji} are the position of $\mathcal{R}_j$ in $\mathcal{F}_{B_i}$. ϕ_{ji}, θ_{ji}, and ψ_{ji} are the roll, pitch, and yaw of $\mathcal{F}_{B_j}$ with respect to $\mathcal{F}_{B_i}$.

The vector can also be expressed in spherical coordinates as

$$\vec{\mathcal{P}}_{ji_{sph}} = \begin{bmatrix} r_{ji} & \beta_{ji} & \alpha_{ji} & \phi_{ji} & \theta_{ji} & \psi_{ji} \end{bmatrix},$$

where: r_{ji} is the *absolute range* between the origins of $\mathcal{F}_{B_i}$ and $\mathcal{F}_{B_j}$, β_{ji} is the *horizontal bearing* of the origin of $\mathcal{F}_{B_j}$ with respect to $\mathcal{F}_{B_i}$, and α_{ji} is the *vertical bearing* of the origin of $\mathcal{F}_{B_j}$ with respect to $\mathcal{F}_{B_i}$.

If ϕ_{ji} and θ_{ji} are dropped under the assumption that the quad-rotors maintain an attitude that is approximately planar, then the framework may be simplified using cylindrical coordinates as

$$\vec{\mathcal{P}}_{ji_{cld}} = \begin{bmatrix} r_{ji} & \beta_{ji} & h_{ji} & \psi_{ji} \end{bmatrix},$$

where h_{ji} is the relative height between the MAVs.

The framework allows to specify a relative pose estimate as one that has all following physical quantities: a) relative range, b) relative horizontal/vertical bearing, c) relative height, and d) relative orientation (yaw).

D-2 Fusion of on-board measurements with environment measurements

The benefits of sensor fusion have already been partially explored in Appendix C, where it described how a gyroscope and magnetometer data are fused to provide a better heading estimate, or when vision is fused with the IMU to differentiate between optical flow due to rotation or due to translation. When using Received Signal Strength Indication (RSSI) as a measure for range, estimates may be poor and subject to unexpected bias and noise depending on the environment, particularly at larger distances. Sensor fusion between relative ranging and relative motion data has been shown to highly increase the estimates that can be provided by these measurements.

Malyavej, Kumkeaw, and Aorpimai (Malyavej et al., 2013) combine IMU and Wireless Local Area Network (WLAN) RSSI measurements using an EKF. With a simple system model (kinematic state equations including only position and velocity) they were able to remove the drift bias from IMU measurements of velocity and position, and ensure position estimates within an error of 1 meter. The on-board IMU measurements were fused with RSS levels from 4 static Wi-Fi APs with known relative positions. Albeit performed in a controlled environment and with multiple static nodes, these kind of results put confidence on the potential improvement of relative localization estimates when combined with on-board odometry measurements.

Similar efforts and results are presented by Rodas, Escudero, Iglesia, et al. in (Rodas et al., 2008), this time using Bluetooth as the Wireless Sensor Network (WSN) of choice. The authors rely on inquiry based RSSI rather than connection based RSSI in order to avoid establishing a paired connection between the devices, which further provides a relevant study on the effect of receiving asynchronous responses. They show that inquiry-only results may be sufficient for low range localization even with long inquiry periods. In simulation, they show the effectiveness of IMU and RSSI fusion over the use of tri-lateration with RSSI, reaching errors below the 1 meter mark in position. The errors only increase when the body accelerates, since the model used by the Particle Filter (PF) does not account for acceleration of an object and needs time to re-converge.

A similar set-up is employed by Subhan, Hasbullah, and Ashraf (Subhan et al., 2013). Their algorithm also uses an EKF based on a kinematic model of an object moving with a fixed velocity in order to smoothen out the Bluetooth RSSI data. Subhan et al. does not include IMU measurements but merely uses the filter to fuse measurements from several beacons.

Pathirana, Ekanayake, and Savkin (Pathirana et al., 2011) also uses a similar approach in order to fuse beacon measurements without IMU.

Raju, Oliveira, and Agrawal (Raju et al., 2012) further observed the benefits of combining RSSI readings with Link Quality Indicator (LQI) as a filter to improve the estimates.

D-2-1 Relative pose measurements and observability

When fusing measurements, un-measured parameters may become observable. This section discusses the contribution that each direct measurement (distance, bearing, orientation, and height) can provide to the observability of the other quantities.

Observability impact of range, bearing and orientation Martinelli and Siegwart (Martinelli & Siegwart, 2005) provides theoretical ground for a discussion on observability perform a Lie observability analysis. A team of two robots is studied in order to compare what bearing-only, range-only, or orientation-only measurements can bring to the observability of the system. It is determined that, fused with odometry measurements, bearing-only and range-only measurements are sufficient to achieve a rank of 3 (i.e. full relative observability) if fused with the odometry measurements. Orientation-only measurements feature a rank of 1 independently of odometry, implying that no state except the relative orientation itself can be observed. Bearing is deemed to provide more valuable information than range, as it is capable of reaching a rank of 3 even when odometry information of both vehicles is not available. The work in (Martinelli & Siegwart, 2005) is complemented by (Martinelli, Pont, & Siegwart, 2005), which is an empirical study (performed in simulation) of the same concept. The superiority of bearing-only measurements over range-only measurements is confirmed. This measurement is capable of maintaining a near-zero error throughout the maneuvers (as opposed to range, for which the location error is seen to diverge slowly). Finally, Sharma, Beard, Taylor, and Quebe in (Sharma et al., 2012) extend the work by Martinelli and Siegwart by analyzing the effect provided by a bearing-only measurement, fused with odometry, in a system of N robots. It is shown that the rank of the *global* system, for which the state vector features $3N$ states, is $3(N-1)$. As expected, full rank of $3N$ is only achievable if static land-marks are also measured.

Degenerate motions with range-only measurements Range measurements have been seen to feature observability issues in two degenerate motions discussed below. For the discussion, consider, for simplicity, the relative localization between two robots $\mathcal{R}_i$ and $\mathcal{R}_j$.

- *Flip ambiguity.* This is an issue that emerges when both robots follow a perfectly straight and mirroring trajectory. It is then impossible, using range-only measurements, to establish on which side of $\mathcal{R}_j$ is with respect to $\mathcal{R}_j$ (Cornejo & Nagpal, 2015) (Zhou, Roumeliotis, et al., 2008). This ambiguity can be resolved by measuring relative orientation and velocity.
- *Rotation ambiguity.* When $\mathcal{R}_j$ perfectly matches the motion of $\mathcal{R}_i$, relative bearing is no longer observable via range-only measurements, as these are constant and provide no useful information. This degenerate motion is unlikely for randomly moving entities (Cornejo & Nagpal, 2015).

Degenerate motions with bearing-only measurements Bearing measurements between two robots allow observability of range only in the case that relative bearing changes over time, which does not happen if two robots are moving along straight and parallel trajectories (with no change in relative orientation either, if measured). It is then not possible to observe the range by only measuring bearing (Mariottini et al., 2009). This issue can be referred to as *range ambiguity*.

Propagation of uncertainty in multi-robot teams Localization can also feature beneficial effects to the overall quality of the estimate as the number of members increases. In fact, the uncertainty in the final estimate is *inversely proportional to the number of agents* and *proportional to the uncertainty of the own-state and relative pose measurements* (Roumeliotis & Rekleitis, 2004) (Sharma & Taylor, 2008). Furthermore, the rate of increase of uncertainty over time does not depend on the uncertainty of the relative pose measurements, but only on that of the uncertainty in the measurement of the own states (Roumeliotis & Rekleitis, 2004). Note that this conclusion is only valid when assuming the case that all robots have access to all other's odometry and pose estimates, and that all relative pose measurements between robots feature the same uncertainty.

D-3 The initial state estimate

An EKF's stability is dependent on the initial state provided (Antonelli, Arrichiello, Chiaverini, & Sukhatme, 2010). If it is insufficiently similar to the actual state, then this could cause the EKF to diverge, which is undesired. It is therefore useful to look into methods to estimate the initial state between the two vectors in order to mitigate the chances of this failure mode. For a 2D multi-robot system, Zhou et al. (Zhou et al., 2008) has proven that 5 range measurements (coupled with odometry) are sufficient to provide an analytical solution for the initial relative pose (relative range, bearing, orientation) between two robots (provided that a degenerate motion as described in Appendix D-2-1 is not taking place). This work has been extended to 6DOF robots operating in 3D in the works (Trawny, Zhou, & Roumeliotis, 2009) and (Zhou, Roumeliotis, et al., 2013), where an analysis for different combinations of measurements of bearing and range over time is also included. It is mathematically shown that no unique solution can exists; featuring at least 2 for a bearing-only measurements scenario, and 40 for a range-only measurements scenario. This problem, which is left untreated in (Zhou et al., 2013) and (Trawny et al., 2009), can be circumvented in this research by allowing the two agents to communicate their attitude with respect to an inertial frame of reference. In this case, both MAVs would have to communicate pitch, roll, and yaw angle to each other.

D-4 Advanced filtering techniques

When models are incorrect and do not match the general trend observed in the measurements, model-based filters can diverge as a result of trying to comply with a faulty/different model. This is the general reason as to why *process noise* is generally included in filters: to account for the inaccuracies that the model may have over the real system (Malyavej et al., 2013).

A series of extensions have been found that can be added to a filter in order to improve its performance and robustness.

D-4-1 Adaptive filters

To avoid divergence with an adaptive method, J. Sasiadek and Hartana (J. Sasiadek & Hartana, 2000) (J. Z. Sasiadek, 2002) suggest the use of weighted noise covariance, which automatically gives higher importance to more recent measurements and releases the filter from the effect of older data. This method, known as *exponentially weighted EKF*, is implemented by updating the process noise covariance matrix $\mathbf{R}$, the measurement noise covariance matrix $\mathbf{Q}$, and the update covariance matrix $\mathbf{P}$ as below, with the weighing parameter $\alpha_w \geq 1$ (note that $\alpha_w = 1$ for standard EKF).

$$\begin{aligned} \mathbf{R_k} &= \mathbf{R}\alpha_w^{-2(k+1)} \\ \mathbf{Q_k} &= \mathbf{Q}\alpha_w^{-2(k+1)} \\ \mathbf{P_k^{\alpha-}} &= \mathbf{P_k^{-}}\alpha_\mathbf{w}^\mathbf{2k} \end{aligned} \tag{D-1}$$

As opposed to selecting an arbitrary α_w, J. Sasiadek and Hartana include a two-input single-output Mamdani fuzzy model to re-evaluate α_w on every iteration. The two inputs are the covariance of the residuals and the mean value of the residuals, and the output is α_w. By using a 3 level semantics of “Zero”, “Small”, and “Large” for both the inputs and the outputs, the Mamdani fuzzy system can be created with only 9 rules represented in a 3×3 table. This augmented EKF is found to be more robust and create smoother estimates than its regular counterpart, and is also more suitable for non-Guassian noise (J. Z. Sasiadek, 2002).

Assa and Janabi-Sharifi (Assa & Janabi-Sharifi, 2015) propose an Iterative Adaptive EKF (IAEKF): a combination of an iterative scheme based on numerical iteration to improve upon the linearization step of EKF, and a recursive adaptation step whereby the process and measurement noise covariance matrices are updated at each step based on the observed differences between the predicted state/original state and the predicted measurement/real measurement over a past horizon of N steps. Albeit the IAEKF seems to come at the cost of increased computation time (experimentally observed between 5 to up to 25 times slower, primarily as a result of the iteration procedure), this algorithm benefits from being more robust to erroneous estimates of the covariance noise matrices and initial state.

D-4-2 Reaching a Consensus

When dealing with a distributed system, a robot $\mathcal{R}_i$ could either determine the state using its own relative measurements of robot $\mathcal{R}_j$, or it could combine its own relative measurements with the reciprocal measurements that robot $\mathcal{R}_j$ is making of $\mathcal{R}_i$ such that they can both reach a final *consensus state*. Within a fusion filter, it is possible to use knowledge about the inherent measurement uncertainty in order to optimize this decision. This is done in the Kalman Consensus Filter (KCF) described by Olfati-Saber in (Olfati-Saber, 2009). The consensus compares the states achieved by the different entities via a consensus gain, similar to a weighted average which is added to the estimate obtained by using the information form of the Kalman Filter (KF), also known as *information filter*. The consensus gain is a function of the update covariance matrix in a the KF.

Appendix E

Literature Review on Collision Avoidance Methods

Path-planning can take two general forms (Fajen & Warren, 2003):

- *global* planning, in which an agent resolves a path based on global knowledge of an environment and its evolution over time, and
- *reactive* planning, where the agent selects actions based only on the current situational knowledge.

The context of obstacle avoidance in an unknown environment lends itself to a reactive planning algorithm, where the intent is to temporarily alter a previously planned trajectory in order to avoid an obstacle. Once the obstacle is avoided, the MAV can return to focus fully on its original goal. Reactive planning is suited to the bottom-up approach to swarm robotics, where the behavior of the system emerges from the individual/independent behavior of all MAVs (Crespi, Galstyan, & Lerman, 2008).

The avoidance reaction can be designed with respect to one or more objectives, such as: avoiding the obstacle within a certain safety margin, minimizing the on-board acceleration efforts during obstacle avoidance, minimizing the change with respect to the original path, minimizing the avoidance execution time (e.g. via an aggressive maneuver). Certain objectives can be seen to be mutually exclusive. There is thus a need for an analysis to determine the desired objectives, and an attempt to gain high-level insights is made by exploring literature related to the human psychology behind obstacle avoidance in Appendix E-1. Appendix E-2 then looks at the aerodynamic issues that may manifest themselves when multiple quad-rotors fly close to each-other. Appendix E-3 explores state of the art practical implementations of obstacle avoidance methods for MAVs and robots. Appendix E-4 explores methods dedicated to directly deal with the uncertainty of a localization estimate.

E-1 Human-driven objectives: the psychology behind obstacle avoidance

Ecological psychology is a branch of psychology dedicated to the understanding of the interactions between an agent and its environment. The underlying principle is that the agent is presented with stimuli, which in turn create opportunities to act (or *affordances*), which can be pursued or not by the agent. Affordances arise and fade as a function of time and of the location of the agent with respect to the environment, and are continuously reassessed by the agent (Araujo, Davids, & Hristovski, 2006). One can see that ecological psychology lends itself to reactive schemes due to its dependence on their immediate surroundings.

E-1-1 Obstacle avoidance

Within the realm of ecological psychology, obstacle avoidance strategies emerge by combining influences from two criteria (Hackney & Cinelli, 2013): 1) *Geometric factors.* These are related to the observed geometry of the obstacle and own geometry. Examples of geometric factors are: distance to the obstacle, bearing of the obstacle with respect to the egocentric reference frame, and (relative) obstacle size. 2) *Dynamic factors.* These involve the relative motion of the agent with respect to the obstacle. Examples include the (estimated) time to impact with the obstacle and the estimated required change in pose/path to avoid it. Dynamic factors may have repercussions on geometric factors, for example by affecting a safety distance between the obstacle and the agent (Hackney & Cinelli, 2013). Schiff and Detwiler (Schiff & Detwiler, 1979), for instance, determined that human subjects have a natural tendency to under-estimate time to collision. This finding is in line with the inclusion of a safety distance in order to account for errors in the estimation of the relative dynamics between the two subjects. Hackney and Cinelli (Hackney & Cinelli, 2013) notes the difference in obstacle avoidance strategies in children, young adults, and adults. Young adults are found to give greater emphasis to geometric factors, possibly due to their greater (perceived) control over the dynamic factors. This hints to the need for a system that is capable of balancing geometric and dynamic factors depending on its dynamic properties/constraints.

Fajen and Warren (Fajen & Warren, 2003) put forward a 2D mathematical model for the control laws of obstacle avoidance in humans. In line with ecological psychology, the model describes the environment surrounding an agent as including *repellers* (obstacles to get away from) and *attractors* (goal locations). The study postulates that the heading-rate ($\dot{\Psi}$) can be described as:

$$\dot{\Psi} = k_g \left(\Psi - \Psi_g\right) + k_o \left(\Psi - \Psi_o\right) e^{-|\Psi - \Psi_o|}. \tag{E-1}$$

This controller ensures that a goal is always relevant unless the agent's heading Ψ is equal to the goal heading Ψ_g, whereas an obstacle becomes relevant if the agent is heading more in its direction (where Ψ_o is the heading towards the obstacle). k_g and k_o are tunable gain parameters. To include the agent's dynamic constraints, this model can be further extended into a 2^{nd} order system with a damping term $-b\dot{\Psi}$, b being the damping coefficient. Furthermore, the effect of goal distance d_g, obstacle distance d_o, turning decay rate with goal distance c_1, acceleration scaling c_2, obstacle scaling c_3, and risk promptness c_4 can been included. The model takes the final form below. It is found to be highly representative of the motion by the human candidates (Fajen & Warren, 2003).

$$\ddot{\Psi} = -b\dot{\Psi} - k_g\left(\Psi - \Psi_g\right)\left(e^{-c_1 d_g} + c_2\right) + k_o\left(\Psi - \Psi_o\right)\left(e^{-c_3|\Psi-\Psi_o|}\right)\left(e^{-c_4 d_g}\right). \tag{E-2}$$

E-1-2 Situation awareness

Attention and expectations may also have an impact on the obstacle avoidance decisions if/when the agent must divert attention towards multiple obstacles simultaneously. In a cluttered scenario (i.e. two or more obstacles), the agent should be able to keep track of more entities. This is a situation that is also likely to occur in a team of MAVs. Ideally, the agent would be able to observe all obstacles with equal accuracy and make an informed escape route that optimizes a given cost function. However, when sensing capabilities are limited (e.g. not omni-directional or featuring lower precision at larger distances), or when processing power is limited, the agent can resolve to balancing high quality pose estimates of only a portion of the obstacles while keeping lower quality pose estimates of other obstacles (Gugerty, 1997).

Situation awareness is defined as the current dynamic model of a given situation that an agent holds. It can be divided into a two-level hierarchy: *explicit* and *implicit*. Explicit refers to high quality information that is ready for use. Implicit refers to lower quality estimates that *can readily become* explicit if needed. Implicit awareness often relies on internal context-based models to keep performing short-term estimates in time-frames when measurements are not available (Gugerty, 1997). This provides a background base for a potential bio-inspired algorithm using multiple layers of cognition. Within the context at hand, this type of breakdown is thought to be useful in the following scenarios:

- If using vision, due to the limited field of view, one could keep models of the movements of MAVs based on the observed initial conditions. These are expected significantly drift over time from reality but, provided sufficiently accurate initial conditions, can be considered reliable in the short term in order to make choices even if the MAV is not detectable in that time-instance. This concept may also be applied to Bluetooth in the case that connectivity intervals are high.
- If computational constraints become an issue due to the need to track multiple entities at once, then the computational efforts can be optimized for focus on the MAVs closer to the agent.

E-2 Aerodynamic considerations for safe swarm operations

MAV teams performing an obstacle avoidance maneuver in 3D environments could have MAVs fly side by side, one below/above the other, or any combination of the two. Depending on the combination, the aerodynamic flow from one MAV can have an unexpected impact on that of the other, which poses a set of constraints on the avoidance maneuver to be selected.

- **Flying side-by-side** Experimental efforts by Powers, Mellinger, Kushleyev, Kothmann, and Kumar, documented in (Powers et al., 2013), have shown that the aerodynamic influence for quad-rotor MAVs is not significant, and the disturbance on the rotor's path is negligible.

- **Flying one above the other** Research by Michael, Mellinger, Lindsey, and Kumar in (Michael et al., 2010) demonstrates that the effect of down-wash can be significant. In such case, the top flying quad-rotor will cause a considerable disturbance to the bottom one, which is undesirable. Furthermore, vertical movements could also cause one quad-rotor to fly too close to the ground, which creates a *ground effect* and pushes the MAV upwards, or fly too close to the ceiling, which creates a pulling effect towards the ceiling (Powers et al., 2013). This implies that vertical maneuvers for collision avoidance are undesirable to avoid unpredictable vertical interactions with both the other MAVs as well as environment features.

Based on the two aforementioned points, it can be concluded that a horizontal maneuver, when possible, is recommended over a vertical maneuver in order to avoid unexpected effects (unless an appropriate controller is used that can aptly take these effects into account, but this is beyond the scope of this research).

E-3 Obstacle avoidance in the context of robotics and MAV multi-rotors

A popular method in robotics for obstacle avoidance when the relative position and velocity of the robots is known is Velocity Obstacle (VO) (Fiorini & Shiller, 1998). The core idea is for a robot to determine a set of all velocities that will lead to collisions, and then choose a velocity outside of that set (usually the one that requires minimum change from the current). VO has stemmed a number of variants (Reciprocal Velocity Obstacle (RVO), Hybrid Reciprocal Velocity Obstacle (HRVO)(Snape, van den Berg, Guy, & Manocha, 2009) (Snape, van den Berg, Guy, & Manocha, 2011), and Optimal Reciprocal Collision Avoidance (ORCA)(Van Den Berg, Guy, Lin, & Manocha, 2011)), which are based on the same concept but further alter the set of forbidden velocities in order to address certain issues such as reciprocity of path smoothness. Another method considered is the Human-Like (HL). These algorithms are deemed to be highly suited to cooperate with the localization scheme at hand because they entirely only on relative pose and velocity. They shall be described and discussed in this section.

In all descriptions below, consider the following case. There is a robot $\mathcal{R}_A$, positioned at a point $\vec{p}_A$ and traveling at speed $\vec{v}_A$, moving towards a robot $\mathcal{R}_B$ at some point $\vec{p}_B$ and traveling at speed $\vec{v}_B$. The robots are represented as circles of radius r_A and r_B centered at $\vec{p}_A$ and $\vec{p}_B$. Furthermore, both wish to travel towards a final goal position, respectively $\vec{p}_{A_{goal}}$ and $\vec{p}_{B_{goal}}$, from which *preferred velocities* $\vec{v}_{A_{pref}}$ and $\vec{v}_{B_{pref}}$ are extracted. The preferred velocities are the velocities that the robots would have if no obstacles were in the way towards its goal position. The two robots are on a collision path, but in the descriptions below $\mathcal{R}_A$ is taken as the protagonist. The schemes are for 2D for robots moving on a plane.

Velocity Obstacle (VO) In this case, $\mathcal{R}_A$ assumes that $\mathcal{R}_B$ is just an obstacle that will not change its velocity due to the collision — there is no sharing of responsibility and it is entirely up to $\mathcal{R}_A$ to perform an evasive maneuver. VO_{AB} is defined as a set of all velocities of $\mathcal{R}_A$ that will lead to a collision with $\mathcal{R}_B$. The set VO_{AB} is calculated based on the collision cone concept (Fiorini & Shiller, 1998). The collision cone CC_{AB} is a cone extending from $\vec{p}_A$ towards $\vec{p}_B$, see left-most figure in Figure E-1. Then:

$$CC_{AB} = \left\{ \vec{v}_{AB} | \lambda_{AB} \cap \hat{B} \neq \emptyset \right\} \tag{E-3}$$

$$VO_{AB} = CC_{AB} \oplus \vec{v}_B \tag{E-4}$$

where: $\vec{v}_{AB}$ is the velocity of $\mathcal{R}_A$ with respect to $\mathcal{R}_B$, λ_{AB} is a line extending from p_A in the direction of v_{AB}, and $\hat{B}$ is a disk of radius $r_A + r_B$ centered around $\vec{p}_B$. Alternatively, as per (Snape et al., 2011), the set can defined as

$$VO_{AB} = \{\vec{v} | \exists t > 0 : (\vec{v} - \vec{v}_B)t \in D(\vec{p}_B - \vec{p}_A, r_A + r_B)\}, \tag{E-5}$$

where: t is time, $D(p, v)$ is a function for a disc centered at $\vec{p}$ of radius r, and $\vec{v}$ is an arbitrary velocity.

The desired evasive velocity $\vec{v}_{A_{des}}$ is then:

$$\vec{v}_{A_{des}} = \underset{\vec{v} \in \{\vec{v} \leq \vec{v}_{A_{max}}\} \cap VO^c}{\arg\min} ||\vec{v} - \vec{v}_{A_{pref}}|| \tag{E-6}$$

It can be easily imagined that if both robots $\mathcal{R}_A$ and $\mathcal{R}_B$ implement VO, then they will both rely on the faulty assumption that the other robot is not going to react to the collision. This will inevitably lead to oscillations (Snape et al., 2011).

Reciprocal Velocity Obstacle (RVO) To fix the oscillations induced by the faulty assumptions when multiple robots all act based on VO, both robots can make the alternate assumption that the other robot will also perform an evasive action — responsibility is fully shared. This uses the same implementation seen for VO, with the only difference that the set RVO_{AB} is used instead of VO_{AB} (Guzzi, Giusti, Gambardella, Di Caro, et al., 2013). The set can be determined as:

$$RVO_{AB} = CC_{AB} \oplus \left(\tfrac{1}{2}\vec{v}_A + \tfrac{1}{2}\vec{v}_B\right) \tag{E-7}$$

RVO solves the problem of reciprocity but leads to a phenomenon coined *reciprocal dances*, which is an initial oscillations that takes place when both robots attempt to cross each-other on the same side. [1]

Hybrid Reciprocal Velocity Obstacle (HRVO) To solve the problem of reciprocal dances and force the collision avoidance algorithm towards a smoother path, HRVO was introduced in (Snape et al., 2011). As the name suggests, this is a hybrid scheme between VO and RVO. A hybrid velocity set is made where the right side uses the RVO boundary and the left side uses the VO boundary. If the velocity is beyond the geometric centerline of the set and into the right side boundary, RVO will be used, otherwise VO will be the choice. When both robots do this, it leads to an increased smooth trajectory and the reciprocal dance problem is eliminated (Snape et al., 2009) (Snape et al., 2011).

[1]It is interesting to note that the concept of reciprocal dance is also observed in humans (Conroy, Bareiss, Beall, & van den Berg, 2014).

Optimal Reciprocal Collision Avoidance (ORCA) Here, set VO^{τ}_{AB} is used instead of VO_{AB}. The sets are similar, but whereas the latter is unbounded in time, VO^{τ}_{AB} is bounded by a time-horizon τ:

$$VO^{\tau}_{AB} = \{\vec{v} | \exists t \in [0, \tau] : \vec{v}t \in D(\vec{p}_B - \vec{p}_A, r_A + r_B)\} \tag{E-8}$$

The required change in velocity u is taken at the boundary of the set (∂VO^{τ}_{AB}).

$$\vec{u} = \underset{\vec{v} \in \partial VO^{\tau}_{AB}}{\arg\min} \; ||\vec{v} - (\vec{v}_A - \vec{v}_B)|| - (\vec{v}_A - \vec{v}_B) \tag{E-9}$$

Each robot is expected to make a change of $\frac{1}{2}\vec{u}$. Note that in ORCA the desired velocity is chosen by minimizing the change with respect to the *current* velocity, as opposed to the preferred one (other options are possible but the above is the best practice as recommended by the authors (Van Den Berg et al., 2011)). ORCA has been successfully tested on real MAVs (Conroy et al., 2014) [2]. Uncertainties and dynamic/kinematic constraints were circumvented by increasing the radii of the MAVs, but reciprocal dances were still observed.

Human-Like (HL) The rationale behind the development of this approach was the achievement of a more human friendly/predictable for robot-obstacle avoidance between humans (Guzzi, Giusti, Gambardella, & Di Caro, 2014). At the core of the HL algorithms there is function $f_o(\Psi)$, which maps each heading direction Ψ [3] to the maximum distance that can be traveled before a collision (Guzzi, Giusti, Gambardella, Theraulaz, et al., 2013). $f_o(\Psi)$ is given an horizon H, such that $f_o(\Psi) \in [0, H]$ (Guzzi et al., 2014). $\vec{c}_o(\Psi)$ is then defined as the point of collision if a given heading Ψ is pursued,

$$\vec{c}_o(\Psi) = f_o(\Psi)\vec{e}(\Psi),$$

where $\vec{e}(\Psi)$ is a unit vector in direction Ψ. The desired heading Ψ_{des} is calculated as

$$\Psi_{des} = \underset{\Psi \in [0, 2\pi]}{\arg\min} \, d\left(s(\Psi), \vec{p}_{goal}\right), \tag{E-10}$$

where $s(\Psi)$ is a segment connecting $\vec{p}_A$ to $\vec{c}_o(\Psi)$ at a given Ψ, and $d(\cdot, \cdot)$ is a function that calculates the minimum distance between a segment and a point. Velocity is chosen as the minimum between the preferred velocity and the velocity needed to reach $f_o(\Psi_{des}$ within a given time.

$$||\vec{v}_{A_{des}}|| = \min\left(||\vec{v}_{A_{max}}||, \frac{f_o(\Psi_{des})}{\tau_1}\right) \tag{E-11}$$

The final result is: $\vec{v}_{A_{des}} = ||\vec{v}_{A_{des}}||\vec{e}(\Psi_{des})$. This change is implemented slowly within the robot in order to smoothen the path in human-like fashion, i.e. $\frac{d\vec{v}}{dt} = \frac{\vec{v}_{des} - \vec{v}}{\tau_2}$. τ_1 and τ_2 are tunable time constants. Analysis of human motion suggest $\tau_1 = \tau_2 = 0.5s$ (Guzzi, Giusti, Gambardella, Theraulaz, et al., 2013). Given the findings of human obstacle avoidance described

[2] On a relevant note: this paper is the closest match at an attempt to fulfill the same aim as the M.Sc. thesis at hand. Conroy et al. implemented a fully on-board relative localization using vision on two Parrot ARDrones 2.0 (with tags), from which they infer relative bearing, distance, and speed. However, the algorithm is only tested and seen to work when the two quad-rotors fly straight towards each-other with forward facing cameras in a controlled environment. The focus of the paper was on the analysis and testing of ORCA in spite of real-life uncertainties and dynamic constraints; it was *not* localization algorithm.

[3] Ψ is not to be confused with ψ, the yaw angle

in Appendix E-1-1, it is unsurprising that this algorithm also focuses on the establishment of a heading, as opposed to directly identifying a velocity as with the VO algorithms and its variants. HL thus presents the advantage that the heading selection is decoupled from speed selection (Guzzi et al., 2014). HL has also been found to be successful even when operating at low rates. Performance is only affected, when the time-step between iterations is larger than $0.4s$ (Guzzi, Giusti, Gambardella, Theraulaz, et al., 2013). This is a very favorable feature for low-speed processors.

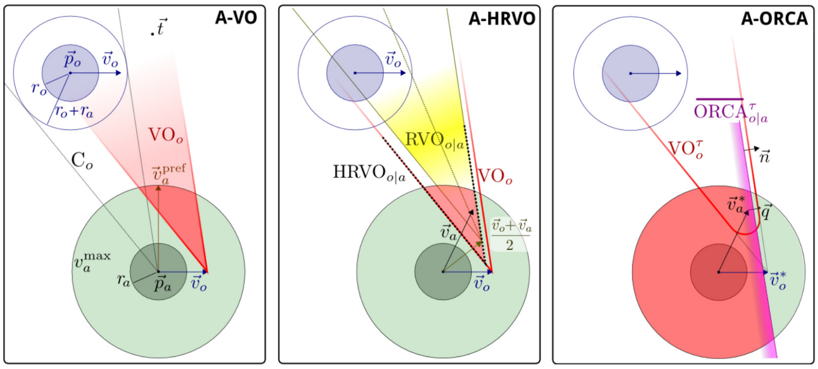


Figure E-1: Representation of VO, RVO, HRVO, ORCA, and HL (Guzzi, Giusti, Gambardella, Di Caro, et al., 2013).

Guzzi, Giusti, Gambardella, Di Caro, et al. (Guzzi, Giusti, Gambardella, Di Caro, et al., 2013) have determined HL to outperform the other methods significantly. However, ORCA has already been tested successfully on MAV quadrotors, and has shown that real-life holonomic constraints as well as uncertainties can be simply included in the algorithm by increasing the perceived radii r_A and r_B of the robots. Unfortunately, despite showing higher performance, HL cannot be proven to always lead to successful collision avoidance. In fact, the authors note that in large crowds collisions between humans are also seen. With a small team of agents, this is not expected to be a problem (Guzzi, Giusti, Gambardella, Theraulaz, et al., 2013).

E-4 Dealing with localization uncertainty

All these algorithms require a precise knowledge of the robot's own state and the obstacle's state in order to work (where state includes pose as well as the change in pose). However, it is undeniable that the final localization estimates will suffer from non-negligible uncertainties. Methods to deal with that are presented in this section.

The geometrical nature of VO-based methods easily lend themselves to dealing with localization uncertainties thanks to the inclusion of an obstacle radii as an integral part of the avoidance scheme. As seen in Figure E-2, the VO_{AB} set can be expanded by the cumulative standard deviation $\omega(\cdot)$ of the localization measurements of both $\mathcal{R}_A$ and $\mathcal{R}_B$, and then further expanded by the uncertainty in the velocity $\vec{v}_B$. The authors suggest that the addition of one standard deviation is sufficient (although this is arbitrary). This concept has been successfully tested on real MAVs performing collision avoidance using ORCA and localizing each-other via vision (Conroy et al., 2014).

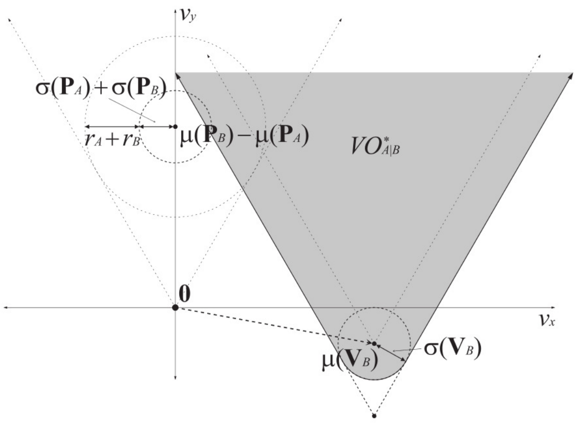


Figure E-2: Geometric expansion of VO according to relative localization and relative velocity uncertainties (Snape et al., 2009).

Another alternative (inspired by human behavior) is that of a *social margin* (Guzzi, Giusti, Gambardella, Theraulaz, et al., 2013) The updated radius is then defined as $r' = r + m(d_s)$, where $m(d_s)$ is a linear function dependent on the absolute distance d_s between the robots. $m(d_s)$ has a positive slope, meaning that the radius is higher at further distances and lower at smaller distances. This brings about two (related) advantages: the likelihood of high-density clusters is reduced, and the robots are more conservative even at larger distances.

The methods described above decrease the chance of collision in-spite of a measurement uncertainty, but do not bind it. Collision Avoidance under Localization Uncertainty (CALU) (Hennes, Claes, Meeussen, & Tuyls, 2012) is a method to do so based on the localization uncertainty. The method is specifically devised for localization via a PF, such that the percentage of enclosed particles can be directly related to the probability of collision according to ORCA. This is a method for ensuring that the radius of the robot is always expanded by an appropriate amount such that the probability of collision stays at some defined constant. The method could be altered for use with an Kalman-type filter.

Appendix F

Preliminary Study of RSSI based Range Measurements between MAVs

The main challenge to be overcome when using RSSI as a distance estimator is the extraction of a sufficiently accurate estimate in spite of a noisy and disturbance-prone environment. The on-board wireless communication system would likely operate in the 2.4GHz frequency spectrum, known as the Industrial, Scientific and Medical (ISM) band. This band is internationally open for use in such applications, and is adopted by a series of popular communication standards such as Bluetooth or Wi-Fi, meaning that a typical modern building can easily become cluttered (Kushki, Plataniotis, & Venetsanopoulos, 2008). Moreover, the signal may be subject to multi-path effects or it can be absorbed from nearby objects (even the human body, for instance, can be a culprit of this) (Kushki et al., 2008) (Caron et al., 2008). Hauert, Leven, Zufferey, and Floreano (Hauert et al., 2010) purposely use 802.11n Wi-Fi transmitting at 5GHz instead of 2.4GHz so as to limit interference from the many devices that operate in the latter. The accuracies are in the order of magnitude of a few meters (Dahlgren & Mahmood, 2014). This improves at closer distances (Nguyen & Luo, 2013) as explained mathematically by the logarithmic Log-Distance path model. For Bluetooth, at distances larger than 5m, RSSI no longer changes and does not contribute to a range measurement.

This section introduces and explores the model used to correlate RSSI with distance. Appendix F-1 introduces the Log-Distance model, which will be central to the work in this thesis. Appendix F-2 performs a sensitivity analysis on the model in order to understand the importance of parameter estimation. Two methods for potentially augmenting the model by means of ground-reflection and inclusion of antenna lobes are treated in Appendix F-3 and Appendix F-4, respectively. The results of an experimental analysis of the model are shown in Appendix F-5.

F-1 The Log Distance path model

The basic model for direct signal loss over an area is the Free Space Loss (FSL) model (Rappaport et al., 1996), which assumes ideal conditions and omni-directional propagation.

The power loss through this model, L_{FSL}, is determined as in Eq. F-1, or as in Eq. F-2 if in dB. Let G_t be the transmitter antenna gain, G_r be the receiver antenna gain, λ_{sig} be the wavelength of the signal [1], and d_s be the absolute range between the two antennas.

$$L_{FSL}(d_s) = G_t \cdot G_r \cdot \left(\frac{\lambda_{sig}}{4\pi d_s}\right)^2 \tag{F-1}$$

$$L_{FSL}(d_s)[dB] = 20\log_{10}\left(\frac{\lambda_{sig}}{4\pi}\right) + 10\log_{10}(G_t) + 10\log_{10}(G_r) - 20\log_{10}(d_s) \tag{F-2}$$

The Log-Distance Path model is based on the FSL model in Eq. F-2, but collapses all constant terms (the first four terms of Eq. F-2) to one value representing the power-loss at a nominal distance of 1m (Rappaport et al., 1996). The path loss using this model, denoted L_{LD}, is given by

$$L_{LD}(d_s) = P_n - 10 \cdot \gamma_l \cdot \log_{10}(d_s), \tag{F-3}$$

where P_n is the total power loss in dB at a nominal distance (1m). Additionally, the coefficient γ_l is included. This is known as the *space-loss parameter*. For free-space $\gamma_l = 2$, showing a resemblance to Eq. F-2. Experimentally, it has been found that office buildings feature $2 \leq \gamma_l \leq 6$ (Kushki et al., 2008). Table 3.2 in (Rappaport et al., 1996), Table 2 in (Ren, Wang, Chen, & Li, 2011), or Table 9.4 n (Seybold, 2005) provide a set of generally recommended values of γ for different environments. The Log-Distance Path model is subject to a Zero-Mean Gaussian Noise (ZMGN) (Rappaport et al., 1996; Svečko, Malajner, & Gleich, 2015)).

The use of the log-distance path model relies on the experimental identification of P_n and γ_l. P_n is a property of the signal and the transmitting/receiving antennas. γ_l is a property of the environment, but as seen in Appendix F-2 its value has a negligible difference on the related distance at small ranges, making it versatile to operate in several environments. Furthermore, the log-distance path model may be augmented by adding the effect of ground reflection or the impact of antenna lobes (for antennas that are not perfectly omni-directional). This is discussed in more depth in Sections F-3 and F-4, respectively.

F-2 Sensitivity Analysis of the Log-Distance Path Model to parameter identification

The use of the log-distance model requires the identification of two parameters, and it is thus worth performing a sensitivity analysis to explore the impact of a parameter on the estimated RSS, and subsequently on the estimated distance. This can help establish how sensitive the identification model is to erroneous parameters or to a change in set-up/environment.

[1]Wavelength can be extracted from frequency thanks to the following relationship: $\lambda_{sig} = c/f_{sig}$, where c is the speed of light and f_{sig} is the frequency of the signal. In this report, the speed of light is assumed to be $3 \cdot 10^8$ m/s.

It is found that P_n is crucial at all distances, as it is directly proportional to the path-loss L_{LD}:

$$\frac{\partial L_{LD}}{\partial P_n} = 1.$$

This is shown in Figure F-1a.

The impact on the distance estimate for the inverted equation is

$$\frac{\partial d_s}{\partial P_n} = ln(10) \cdot \frac{1}{10\gamma_l} \cdot 10^{\frac{P_n - L}{10\gamma_l}},$$

showing that the impact is aggravated as the absolute difference between the actual power loss and P_n increases. This effect is depicted in Figure F-1b.

The impact of a change of γ_l is less influential. This is seen by extracting its derivative with L_{LD}:

$$\frac{\partial L_{LD}}{\partial \gamma_l} = -10log_{10}(d_s). \tag{F-4}$$

If $d_s = 1m$, then a change in γ_l has no influence. At $d_s = 3m$, a reasonable boundary within the scope of this project, then $\Delta L_{LD}/\Delta\gamma_l = 6.02dB$. This is a less significant impact compared to the one of P_n. The effects are plotted in Figure F-2a.

Finally, the impact of γ_l on the distance estimate when the log-distance model is inverted is given by:

$$\frac{\partial d_s}{\partial \gamma_l} = -ln(10) \cdot \underbrace{10^{\frac{P_n - L}{10\gamma_l}}}_{Term\ \#1} \cdot \underbrace{\frac{P_n - L}{10\gamma_l^2}}_{Term\ \#2}. \tag{F-5}$$

At smaller distances, where $P_n - L >> 0$ ($\therefore\ d << 1$), the impact of a change in γ_l is dominated by *Term #2*, which has minimal impact in terms of absolute numbers. At larger distances, where $P_n - L << 0 \therefore d >> 1$, then the change is dominated by *Term #1*, which grows exponentially and begins to become relevant. This is plotted in Figure F-2b.

Overall, it can be concluded that the estimate of P_n needs to have a good accuracy in order to avoid systematic errors at all distances. For the distances expected within the scope of this project ($0m \leq d_s \leq 3m$), a proper identification of γ_l is less important and will have a smaller impact. As P_n encloses terms related to antenna gain, this value is expected to be mostly dependent on the set-up antennas used. γ_l, alternatively, is dependent on the environment. This means that it should be possible to export the same set-up to a different environment with minimal impact (at small distances).

F-3 Possible model augmentation for ground-reflection

The signal from an antenna will reflect on the surrounding elements of the environment, and this reflection may cause multi-path interference with the original signal causing either an increase (constructive interference) or a decrease (destructive interference) of the RSS. If flying indoors in an unknown environment, the only predictable element that can be modeled is the floor/ground.

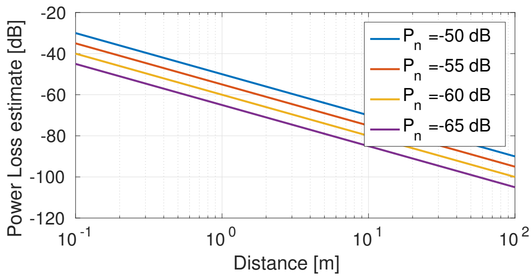


Figure F-1a: Sensitivity analysis of changing P_n in log-distance model on the estimated power loss at different distances. $\gamma_l = 2$.

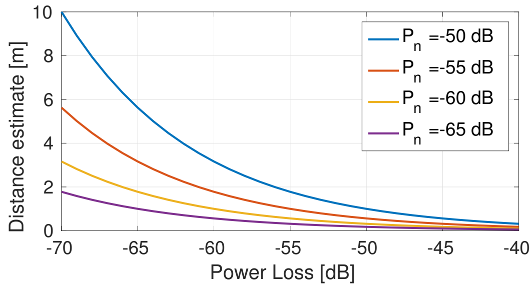


Figure F-1b: Sensitivity analysis of changing P_n in log-distance model on the estimated distance at different power losses. $\gamma_l = 2$.

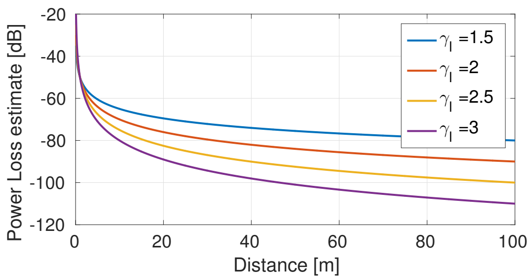


Figure F-2a: Sensitivity analysis of changing P_n in log-distance model on the estimated power loss at different distances. $P_n = -50$.

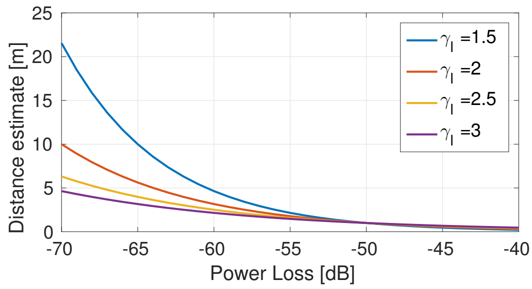


Figure F-2b: Sensitivity analysis of changing P_n in log-distance model on the estimated distance at different power losses. $P_n = -50$.

The effects of multi-path due to the ground can be modeled using the *two-ray ground reflection model*, or *ground model* Goldsmith (Goldsmith, 2005), Svečko et al. (Svečko et al., 2015), and Seybold (Seybold, 2005). The ground model is based on the diagram in Figure F-3; the equations presented in this chapter all make use of the symbols therein.

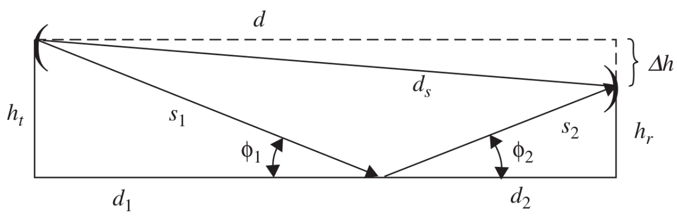


Figure F-3: Depiction of signal propagation over ground assuming perfect reflection. This is the and basis for the two-ray ground reflection model (Seybold, 2005). The transmitter is seen on the left and the receiver is seen on the right.

In Figure F-3, note that $\Phi_1 = \Phi_2$, and d is the *planar* distance between the antennas.

The phase difference $\Delta\Theta$ can be calculated with Eq. F-6 (Seybold, 2005). Note that this uses the key assumption that the ground is a perfectly smooth and reflecting surface from which the signal is reflected with equal magnitude and opposite phase.

$$\Delta\Theta = d\left(\sqrt{1+\frac{(h_r+h_t)^2}{d^2}} - \sqrt{1+\frac{(h_t-h_r)^2}{d^2}}\right)\frac{2\pi}{\lambda_{sig}} \tag{F-6}$$

Furthermore, the total distance s that the reflected signal will travel is (Seybold, 2005):

$$s = s_1 + s_2 = (h_t + h_r)\sqrt{1 + \frac{d^2}{(h_t + h_r)^2}} \tag{F-7}$$

The ground-reflection model at short distances is then given as in Eq. F-8, as a function of the direct propagation distance d_s and the reflected distance s (Baunach, Mühlberger, Appold, Schröder, & Füller, 2009).

$$L_{FSL,ground}(d_s, s) = L_{FSL}(d_s) + L_{FSL}(s) \cdot cos(\Delta\Theta) \tag{F-8}$$

If expressed in dB in order to augment the Log-Distance Path model, then:

$$L_{LD,ground}(d_s, s) = L_{LD}(d_s) + 10log_{10}\left(1 + \frac{L_{FSL}(s) \cdot \cos(\Delta\Theta)}{L_{FSL}(d)}\right) \tag{F-9}$$

Although based on the same concept, this model differs from the models suggested by Goldsmith Goldsmith or Seybold (Seybold, 2005), which may be more commonly seen in literature. It is better suited to the propagation of a signal at short distances by not making the assumption that $d_s >> h_t, h_r$ or that $d_s \approx s$.

Experimentally, this behavior is confirmed by (Nguyen & Luo, 2013), for which it was found that multi-path (ground) effects are most dominant at distances larger than $2m$ and only noticeable at heights closer to the ground. This is modeled mathematically in the model by the fact that as $d_s \to 0$, then $d_s << s$ and the ratio of $L_{FSL}(s)/L_{FSL}(d) \to 0$.

Based on the model in Eq. F-8, the impact of ground reflection for a 2.4GHz (Bluetooth/Wi-Fi) signal is shown in Figure F-4 for the cases where $h_t = h_r = 1m$, $h_t = h_r = 2m$, and $h_t = h_r = 3m$ was tested. The following two behaviors are identified:

- The effect is significant when both antennas are at 1m, but decreases significantly at larger heights.
- The likely-hood of entering regions of full constructive and destructive interference increases with height. This may be seen as an advantage (decreased time for which the effect takes place if the MAV is moving), or a disadvantage (data will feature higher frequency noise/disturbances).

F-4 Possible model augmentation with antenna lobes

Although all models in previous examples assume omni-directional propagation in all directions, this is not necessarily the case for Bluetooth antennas, which may feature antenna lobes (Nguyen & Luo, 2013). Antenna lobes are specific to each antenna (Nguyen & Luo, 2013), unless an assumption is made regarding the fact that each antenna made from a manufacturer will feature the same lobe distribution. Experimentally, Nguyen and Luo found that RSS readings are susceptible to sensor orientation giving a change of up to 10 dBm (Nguyen &

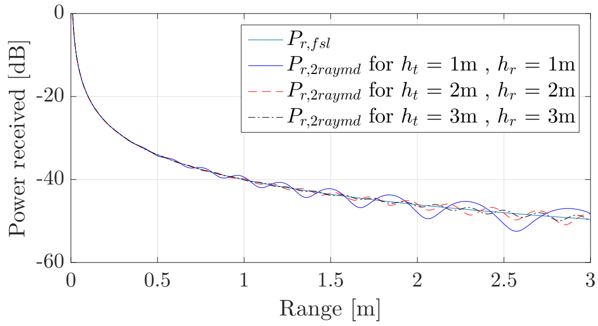


Figure F-4: Difference between free space loss and ground reflection model applied to a Bluetooth type signal (2.4 Ghz). Height of transmitter (h_t) and receiver (h_r) is 1m, 2m, and 3m. The speed of light is assumed as $3 \cdot 10^8$ m/s.

Luo, 2013). Lobes may be identified based on empirical data, this is done in Appendix F-5-3, but it should be noted that the electronics on an MAV could alter the lobe shape further, meaning that occasional re-calibration may be needed.

In the Log-Distance Path model, lobes may be included as an *additional gain term* that models the deviation that the lobe causes from the average value P_n at a distance of $1m$. The lobe impact is included as a non-linear function $\mathcal{G}$ of horizontal bearing β and vertical bearing α of the receiving antenna with respect to the transmitting antenna (denoted by subscript rt).

$$G_{r_{lobe}} = \mathcal{G}(\beta_{rt}, \alpha_{rt}) \tag{F-10}$$

F-5 Experimental Results using Bluetooth RSSI with the Ladybird MAV

Lisa-S Ladybird drones already have Bluetooth connectivity, and therefore sample measurements to test the effectiveness of Bluetooth RSSI as a distance estimation method at short range have been performed, using an Optitrack system in order to also measure the ground-truth position. All experiments featured a static near omni-directional Bluetooth antenna (W1049B by Pulse) communicating with the Bluetooth antenna on a moving Ladybird MAV.

This section showcases representative results of the experiments that have been performed, and studies the performance of the log-distance model in Eq. F-3 as a means to represent the system. Future iterations of the model, outside of the scope of this report, will also include the effects of antenna lobes and ground-path effects.

F-5-1 Description of experiments

The performed tests with relevant data used in this section are described below. They are referred to by the day in which they have been performed (the experiments were performed over a total of three days), and the log of the experiment. [2]

[2]Some logs are not included here as they do not have information that is relevant for this report.

- **Day 1, Log 1 (β constant, hand-held)** — The MAV was held such that the antenna would be with constant bearing with respect to it, and concentric circles around the antenna were walked at radii of $\approx$ 1m, 2m, and 3m.
- **Day 1, Log 2 (β varying, hand-held)** — The MAV was held at constant heading such that the bearing of the antenna with respect to it would vary as it was moved in and concentric circles around the antenna at radii of $\approx$ 1m, 2m, and 3m.
- **Day 2, Log 2 (β varying, α constant, hand-held)** — The MAV was held such that the antenna would be with constant bearing with respect to it, and concentric circles around the antenna were walked at radii of $\approx$ 1m, 2m, and 3m. At several steps during the concentric circles, the MAV was rotated about its y_B axis to change its pitch.
- **Day 2, Log 3 (β varying, α varying, hand-held)** — The MAV was held at constant heading such that the bearing of the antenna with respect to it would vary as it was moved in and concentric circles around the antenna at radii of $\approx$ 1m, 2m, and 3m. At several steps during the concentric circles, the MAV was rotated about its y_B axis to change its pitch.
- **Day 3, Log 1 ($h_t = h_r \approx 1m$, in-flight)** — The MAV was made to fly in 3 concentric circles, at radii of 1 m, 1.75 m, and 2.5 m around an antenna. Both were approximately at 1 m height from the ground. The heading was set to remain approximately constant during the flight.
- **Day 3, Log 2 ($h_t = h_r \approx 2m$, in-flight)** — The MAV was made to fly in 3 concentric circles, at radii of 1 m, 1.75 m, and 2.5 m around an antenna. Both were approximately at 2 m height from the ground. The heading was set to remain approximately constant during the flight.
- **Day 3, Log 3 ($h_t = h_r \approx 3m$, in-flight)** — The MAV was made to fly in 3 concentric circles, at radii of 1 m, 1.75 m, and 2.5 m around an antenna. Both were approximately at 3 m height from the ground. The heading was set to remain approximately constant during the flight.

Furthermore, it is worth noting the following:

- All measurements taken on the same day were performed shortly after each-other. Environmental factors and disturbances are assumed constant during any experiment performed on the same day.
- Experiments on day 1 and day 2 were performed using the same antenna and the same MAV. The experiments on day 3 were performed using another set. Using a different antenna can be expected to bring about significant changes in P_n.

F-5-2 Fitting the log-distance model

In the log-distance model, it is necessary to identify the nominal parameter P_n and the space-loss parameter γ_l that best match the given conditions. This has been done for all experiments detailed in the previous subsection (Appendix F-5-1) using a non-linear identification methods that aimed to minimize the difference between the model estimate and the measured values (using command `fmincon` on Matlab).

Table F-1a shows the results of the parameter identification based on a randomly selected 10% of the data set. Table F-1b show the parameter estimation based on the full data set. It can be seen that the values between the two tables are close, showing that a low amount of datapoints is sufficient to identify the model parameters. The largest differences are seen with the experiments of Day 3, where γ_l features a significant variation between the two tables. This is explained by the fact that a portion of the measurements (prior to take-off or after landing) featured the MAV being placed on the ground, which can enhance the multi-path or disturbances experienced. These contouring measurements have not been deleted from the data-set. It may also be seen that the values estimated for the experiments of day 3 feature a lower value of P_n. At this stage, the cause for this is unclear. It may be due to a different antenna/MAV set or due to environmental disturbances during the flight experiment.

Table F-1a: Estimated parameters of log-distance model (Eq. F-3) from all performed experiments using randomly sampled 10% of data.

	$P_n[dB]$	$\gamma_l[-]$
Day1, Log 1	-50.797	1.889
Day1, Log 2	-52.382	2.334
Day2, Log 2	-55.870	1.821
Day2, Log 3	-57.058	1.907
Day3, Log 1	-64.222	1.256
Day3, Log 2	-62.865	2.331
Day3, Log 3	-62.445	2.625

Table F-1b: Estimated parameters of log-distance model (Eq. F-3) from all performed experiments using the full data sets.

	$P_n[dB]$	$\gamma_l[-]$
Day1, Log 1	-51.501	1.880
Day1, Log 2	-53.312	2.265
Day2, Log 2	-56.122	1.761
Day2, Log 3	-57.382	1.796
Day3, Log 1	-63.727	1.400
Day3, Log 2	-63.182	2.453
Day3, Log 3	-63.480	2.566

The estimated parameters (using 10% of the data set) can be entered on the model and used to determine an estimate of power loss (equivalent to RSSI in this context), or of distance d if Appendix F-1 is inverted. The results of this are shown in Figure F-5a to F-5b for *"Day 1, Log 1"*, Figure F-6a to F-6b for *"Day 2, Log 2"*, and Figure F-7a to F-7b for *"Day 3, Log 1"*.

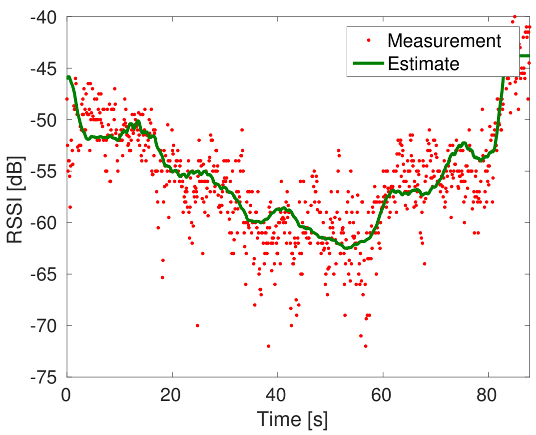


Figure F-5a: Raw RSSI measurements of *"Day 1, Log 1"* against log-distance Power Loss estimate with relevant parameters from Table F-1a and measured distance.

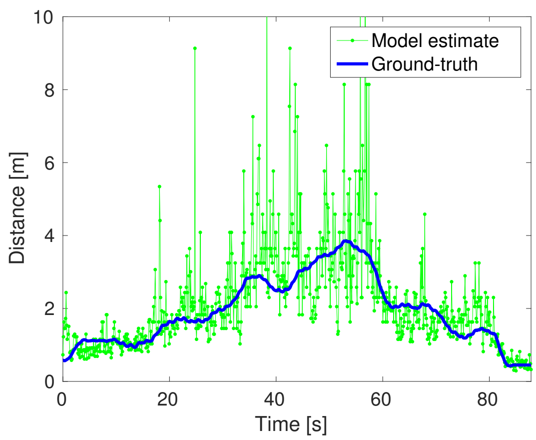


Figure F-5b: Actual distance of *"Day 1, Log 1"* against inverted log-distance distance estimate with relevant parameters from Table F-1a and measured RSSI.

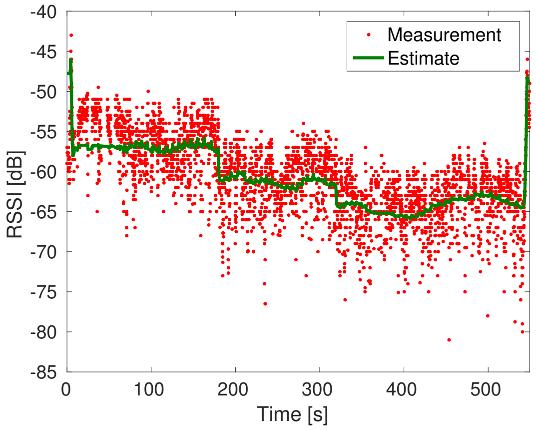


Figure F-6a: Raw RSSI measurements of *"Day 2, Log 2"* against log-distance Power Loss estimate with relevant parameters from Table F-1a and measured distance.

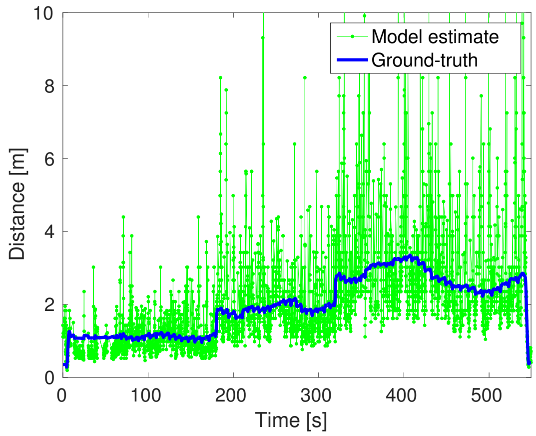


Figure F-6b: Actual distance of *"Day 2, Log 2"* against inverted log-distance distance estimate with relevant parameters from Table F-1a and measured RSSI.

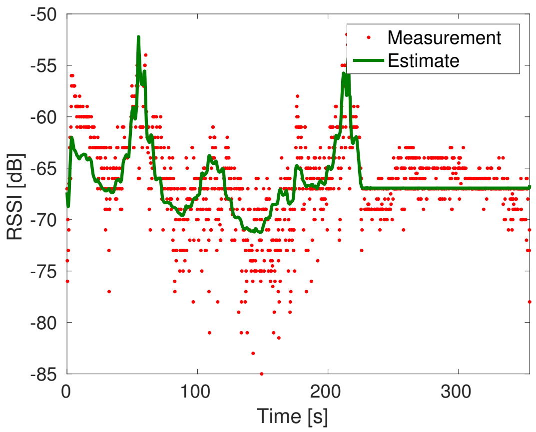


Figure F-7a: Raw RSSI measurements of *"Day 3, Log 1"* against log-distance Power Loss estimate with relevant parameters from Table F-1a and measured distance.

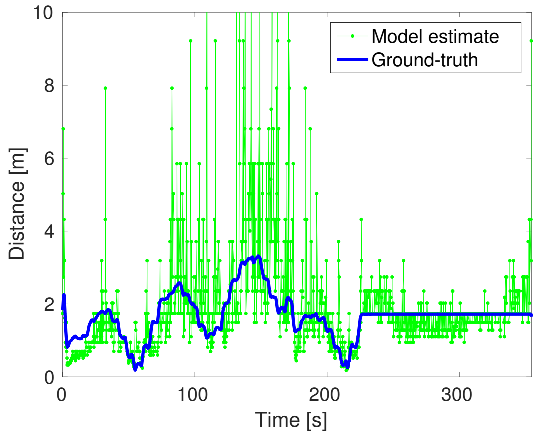


Figure F-7b: Actual distance of *"Day 3, Log 1"* against inverted log-distance distance estimate with relevant parameters from Table F-1a and measured RSSI.

The error was calculated as the difference between the measurement and the estimate. Its distribution is shown in Figure F-8a and Figure F-8b for the RSSI estimates and the distance estimates, respectively. Whereas the error distribution in the RSSI estimate appears (almost) normal, as expected based on (Rappaport et al., 1996), it can immediately be recognized that the error features a Gamma/Log-normal/Nakagami type distribution when converted to distance (this was also expected based on the findings by (Seybold, 2005) and (Svečko et al., 2015)). The error distribution has a sharp peak and slower descent on the negative side. It is also worth noting that the peak is generally higher for experiments where antenna lobes and height effects play a less significant role (with the peak for *Day 1, Log 1* being the highest). The RMSE errors over the full data sets for all experiments are shown in Table F-2. RMSE is best used for normal/Gaussian distributions. This may not be directly applicable to the noise in the distance estimates, but it is worth mentioning as an indication for the quality of the estimates, showing that the largest RMSE is $\approx 2.5m$.

Table F-2: RMSE of model fits (from model with parameters extracted from 10% of data points)

	RMSE RSSI estimate [dB]	RMSE distance estimate [m]
Day1, Log 1	3.139	1.277
Day1, Log 2	5.880	2.682
Day2, Log 2	3.510	1.418
Day2, Log 3	4.948	2.360
Day3, Log 1	3.204	2.443
Day3, Log 2	6.033	2.232
Day3, Log 3	4.969	1.395

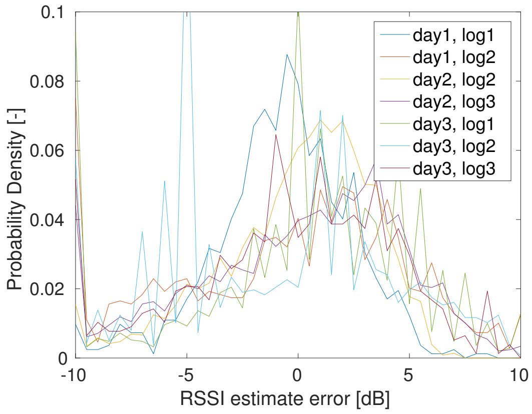


Figure F-8a: Error distribution between the measured RSSI and the estimated RSSI (from relevant parameters in Table F-1a from 10% of data).

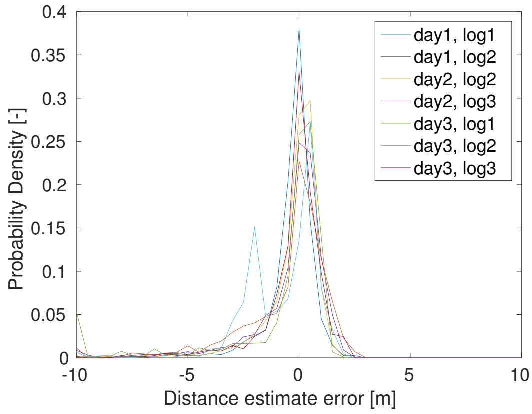


Figure F-8b: Error distribution between the real range (as from the Optitrack measurement) against the estimated range based on the inverted Log-Distance model (with relevant parameters in Table F-1a from 10% of data).

F-5-3 Antenna lobe investigation

As noted in Appendix F-5-1, certain tests keep the bearing of the omni-directional antenna with respect to the MAV antenna constant, whereas others did not. The impact of a lobe distribution for the MAVs in question is explored in this section, where the log-distance model is augmented with a term describing the non-linear lobe patterns, as follows:

$$L_{LD} = P_n - 10 \cdot \gamma_l \cdot \log_{10}\left(\frac{d_s}{d_0}\right) + G_{r_{lobe}} + N_p, \tag{F-11}$$

This section aims to validate the claims found in (Nguyen & Luo, 2013) by establishing whether there are systematic differences due to bearing and to assessing their impact. Note that the lobes in this subsection identified using all data points within a measurement *and not* only 10% as was done previously. This is because the focus is on identifying the shape of the lobes with the highest accuracy possible. For lobes on the 2D plane, $G_{r_{lobe}} = \mathcal{G}(\beta_{tr})$. The function is expressed using 3^{rd} degree splines, and the parameters of the splines are identified simultaneously with P_n and γ_l. This implementation of the spline lobe identification was based on the work by Szabo in (Szabo, 2015). For 3D lobes, where $G_{r_{lobe}} = \mathcal{G}(\beta_{tr}, \alpha_{tr})$, the function is currently described as a crisp set (i.e. a look-up table). The impact of the lobe is obtained by performing sectioned averaging in order to determine the average impact of a lobe over sections of combinations of horizontal and vertical bearings. Eq. F-11 only contains one term for the antenna lobes, albeit there would have to be two (one for each antenna). As one of the antennas used in all experiments was omni-directional, this is ignored for now.

As a first step, experiments *"Day 1, Log 1"* and *"Day 2, Log 2"* were compared. In *"Day 1, Log 1"*, for which the raw results are shown in Figure F-9a to F-9c, the relative bearing of the omni-directional antenna with respect to the drone was kept constant, and the bearing of the MAV with respect to the omni directional antenna was changing as concentric circles were made. A truly omni-directional antenna should not show any systematic changes in RSSI due to bearing, and this was confirmed by the measurements in Figure F-9c, from which no impact of bearing higher than 1dB can be extracted. In *"Day 1, Log 2"*, with the measurements shown in Figure F-10a to F-10c, it is possible to qualitatively notice a systematic variation in RSSI as a result of relative bearing (systematically higher at ≈ 1 rad). The results of the parameter identification combined with the spline lobe estimation are presented in Figure F-11d, from which it is possible to observe lobes featuring an impact of up to $-7dB$ on the omni-directional signal.

To analyze the 3D lobe distribution, the results of the hand-held test *"Day 2, Log 3"* were used. This is because this data set features the highest amount of combinations for relative horizontal and vertical bearings between the antennas. The raw data of the measurements is shown in Figure F-11a to F-11c, from which a specific pattern as a function of bearing is clearly noticeable merely from the graphs. Based on the identification procedure, the maximum observed impact of the lobes reaches $\approx -10dB$. The mean impact is of $-3dB$. These results are in line with the expectations in (Nguyen & Luo, 2013), which also expresses $10dB$ as a likely maximum level of impact of the lobes on Bluetooth antennas.

It is established that the antennas on the MAVs feature systematic lobe effects, and an accurate modeling of these lobes may be helpful towards improving a distance estimate. Based on the findings in the sensitivity analysis of the log-distance model, a lobe distribution will have the same impact on the modeled RSSI as a change in P_n, i.e. directly proportional. As the worst cases show effects up to $10dB$, this is not to be neglected. Unfortunately, regardless of the accuracy of the modeling, a naive implementation of this model in a localization scheme can lead to a circular issue whereby the relative bearing needs to be established in order to improve the measurement of the bearing, and vice-versa. If this type of modeling is explored further for implementation, then it would be necessary to establish a method to pre-estimate the relative bearing. A current (yet to develop) idea to do so is as follows. As noise in RSSI was determined to be of near-Gaussian nature with zero mean, one could attribute a case where the estimate is consistently offset to lobe. For instance, if there is a long period over which the RSSI is consistently higher than the predicted value, then this can be taken as being due to a lobe, which can condition the algorithm towards a certain bearing. The danger in doing so is that this may be founded on a wrong assumption, as the consistent change over a longer period may be due to other factors such as multi-path effects or other disturbances in the environments. Alternatively, the methods of Zhou et al. (Zhou et al., 2013) and Zhou et al. (Zhou et al., 2008) could be adopted in order to reach initial estimates for relative bearing

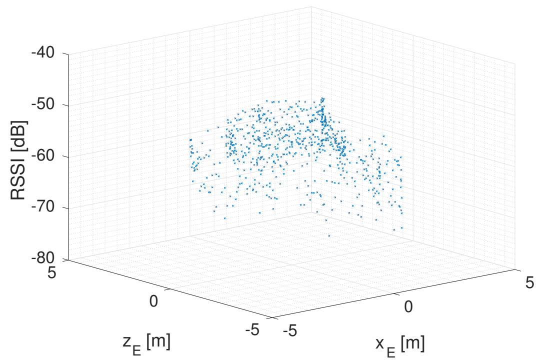


Figure F-9a: Planar location in inertial reference frame $\mathcal{F}_E$ with RSSI values recorded by receiver in experiment *"Day 1, Log 1"*.

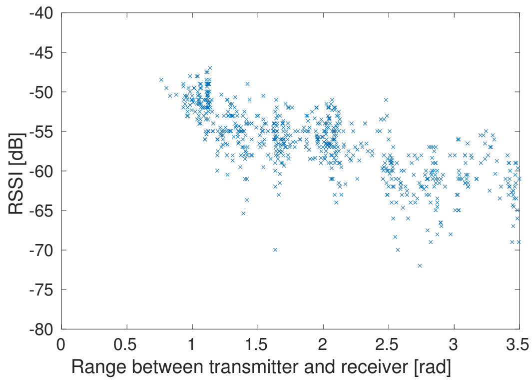


Figure F-9b: Range against RSSI levels of experiment *"Day 1, Log 1"*.

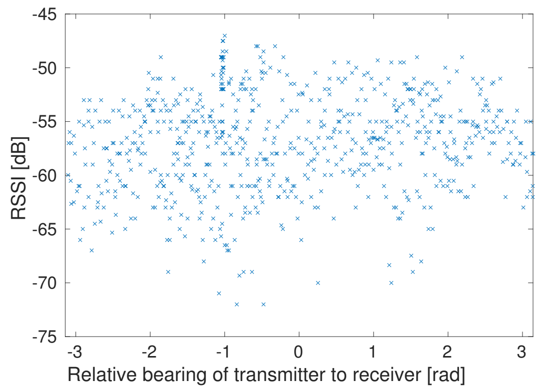


Figure F-9c: Relative bearing against RSSI levels of experiment *"Day 1, Log 1"*.

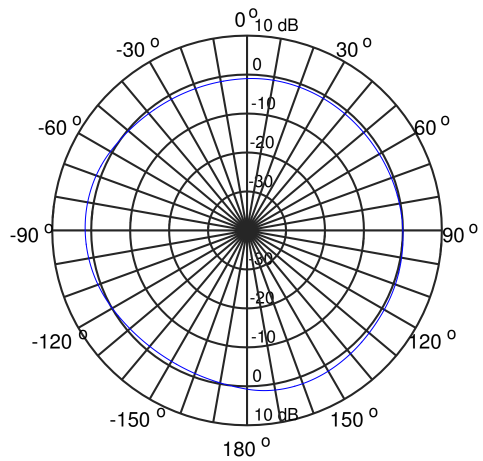


Figure F-9d: Results of parameter identification with horizontal lobe identification for the omni-directional antenna using data of experiment *"Day 1, Log 1"*.

that can then be improved. These methods are discussed in more detail in Appendix D-3.

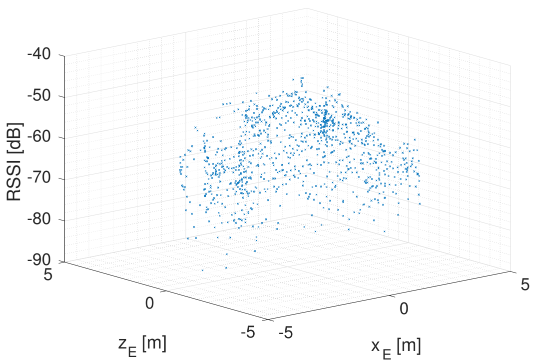


Figure F-10a: Planar location in inertial reference frame $\mathcal{F}_E$ with RSSI values recorded by receiver in experiment *"Day 1, Log 2"*.

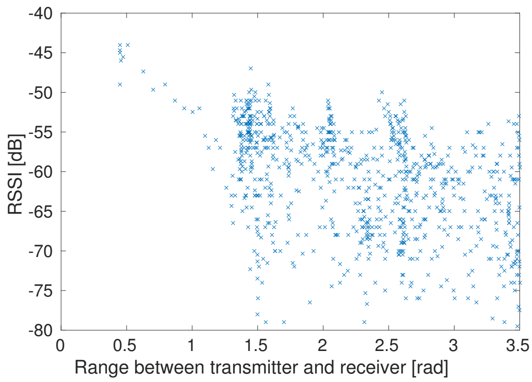


Figure F-10b: Range against RSSI levels of experiment *"Day 1, Log 2"*.

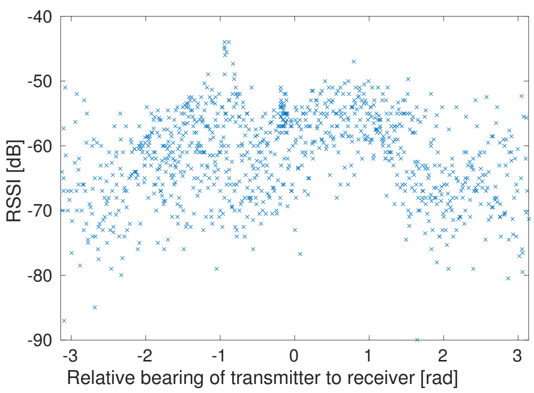


Figure F-10c: Relative bearing against RSSI levels of experiment *"Day 1, Log 2"*.

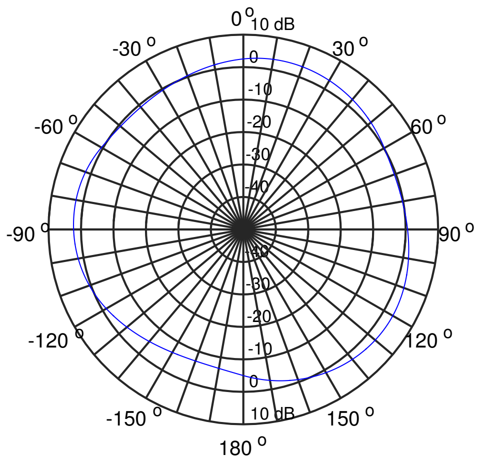


Figure F-10d: Results of parameter identification with horizontal lobe identification for MAV antenna using data of experiment *"Day 1, Log 2"*.

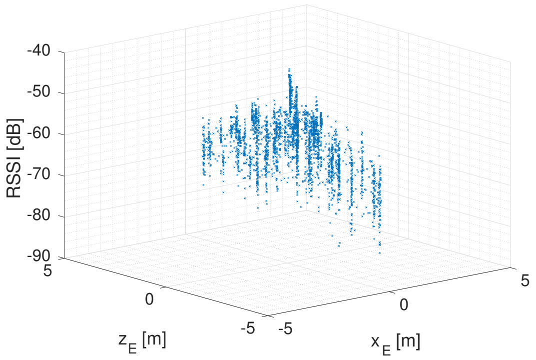


Figure F-11a: Planar location in inertial reference frame $\mathcal{F}_E$ with RSSI values recorded by receiver in experiment *"Day 2, Log 3"*.

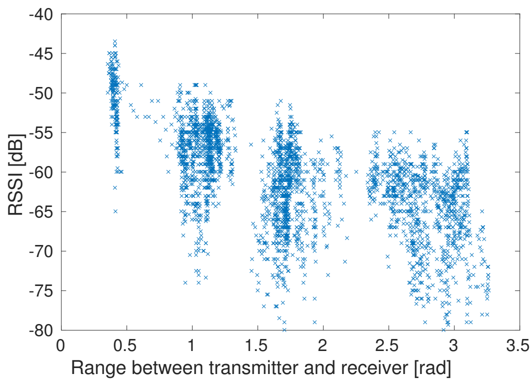


Figure F-11b: Range against RSSI levels of experiment *"Day 2, Log 3"*.

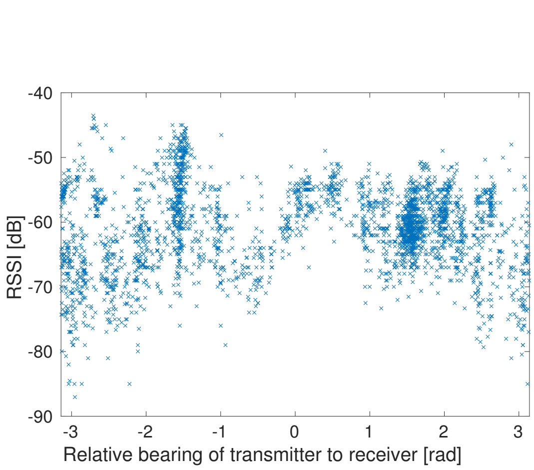


Figure F-11c: Relative bearing against RSSI levels of experiment *"Day 2, Log 3"*.

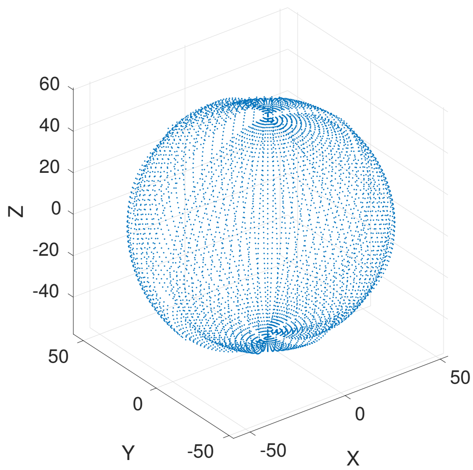


Figure F-11d: Results of parameter identification with full lobe identification for MAV antenna using data of experiment *"Day 2, Log 3"*.

Appendix G

Alternative Study of Vision as a Measure for Relative Range/Bearing

Vision-based methods use a camera in order to infer properties about the environment. In Appendix C it was seen that this is useful to estimate the state of the MAV with respect to the environment. This section details the possibilities of using this sensor in order to detect nearby MAVs. If the object is properly recognized, it is then possible to extract the relative bearing. Range can also be extracted either by observing its size compared to a reference or by the use of a stereo camera. Finally, by observing the objects one could even extract the relative *orientation/attitude*, albeit this last point is unexplored.

G-1 Literature review

G-1-1 Direct detection of other MAVs using vision

To simplify the visualization tasks, there are several instances where extra visual cues are added to the quad-rotors in order to simplify the job of identification. This generally takes the form of one or more colored tags placed on the object. Iyer, Rayas, and Bennett (Iyer et al., 2013), for instance, use rectangular orange tags attached to each side of a Parrot AR drone v1.0 in order to make the drone more visible to the camera (and providing the simpler task of detecting rectangular shapes). The separation and relative orientation between the tags are used to also estimate the distance to the other drones. Using a similar strategy but with a different execution, Chi Mak, Whitty, and Furukawa (Chi Mak et al., 2008) use two cyan Light-Emitting Diodes(LEDs) oppositely placed on the end of the main rotor of a helicopter MAV, combined with one red LED at the tail end. The effect, when the rotor is spinning, is a cyan ellipse of which the size, inclination, and relative brightness can give an indication of relative pose. This solution requires that no other cyan or red objects are visible in the footage. Furthermore, to ensure an ellipse is fully captured, the camera is forced to operate at a rate slightly lower than half of a rotor revolution. A filter is then needed over the main lens in order to avoid potential over-exposure from other light-sources. The system is found to be accurate but not robust to changes in yaw (which may hide the red LED) or range.

The aforementioned strategies are not deemed applicable to the given scenario if the focus is restricted to swarm collision avoidance. (Iyer et al., 2013) and (Chi Mak et al., 2008) cannot be used because LEDs and tags cannot be attached to the MAVs and the solution should aim to be independent of auxiliary markers and visual cues which must be known a-priori in order to work. If no simple visual cues are offered, then objects must be recognized and tracked based on detectable features. Three popular methods for object recognition, also discussed in more detail Appendix B-5, are introduced below.

- **Template matching**: A simple example is given in (Corke, 2011), used to "find Waldo" in a cluttered picture. Despite low quality template, the algorithm is successful. Template matches is not robust to changes in perspective.
- **Line features**: Through an edge image, line features can be detected which can be matched to an internal edge model of an object for object recognition as in (Lowe, 1987).
- **Point features**: (Invariant) point features, memorized prior to flight, can be used to recognize an object such as an obstacle (Lowe, 1999).

Alternatively, optical flow can be used for obstacle detection and recognition as discussed in Appendix G-1-3.

G-1-2 Relative localization via shared environment data

Relative localization can also take place without the need for the members to directly detecting each-other, but by sharing their cognition of environment features. The key concept here is to allow a team of robots to detect and localize themselves with respect to one or more shared object(s), after which they can infer their relative location.

Lima, Santos, Oliveira, Ahmad, and Santos (Lima et al., 2011), in the context of soccer-playing robots, use range localization relative to a red ball in order to establish a distance to it. Based on this, when a team-member is "lost" (does not know where it is in the field), it can ask for assistance from the other robots, at which point the range of each robot to the red ball is received. As all the other robots know their global position in the field, the lost robot can re-determine its position in the global frame (fusing all data with a particle filter).

Saska (Saska, 2015) provides swarming capabilities to a swarm of quad-rotor MAVs using only a monocular camera on each. Although the system does not require external sensors, it does require for circular markers to be set in known locations. The markers feature two rings, which are simple to locate and track, and are of known dimensions.

On a similar note, Kendall, Salvapantula, Stol, et al. (Kendall et al., 2014) rely on a round colored targets in order to infer the location to an object. These targets are recognized by using a blob-detector (using a template matching method), in order to extract a distance estimate and control a quad-rotor.

These research papers make use of the benefit of having pre-established, easily recognizable target objects to identify in an environment. Furthermore, all implementation are active — the robots *seek* the target object and it is thus supposed to be in its field of view, which may otherwise not be the case and result in significant problems (this issue is discussed further based on experimental results in Appendix G-2-5).

G-1-3 Obstacle detection using optical flow

Optical flow can also provide a basis for obstacle detection by observing the difference among sections. When one MAV is flying in the camera view, the optical flow should show one element (i.e. the MAV) moving differently from the rest of the environment (assumed static). This motion implies the presence of an MAV. Alternatively, the use of *textons* in optical flow has also shown success for obstacle recognition in landing (Ho, De Wagter, Remes, & de Croon, 2015) and normal flight (De Croon, De Weerdt, et al., 2012). Ho et al. (Ho et al., 2015) used a learning algorithm that allocates a texture to an optical flow parameter, which, if inconsistent with the plane, is judged to be an obstacle and is later recognized as one. Another example in this context is by De Croon, De Weerdt, et al. (De Croon, De Weerdt, et al., 2012), where obstacle detection in normal flight with a forward facing camera was performed by observing the change in texture of the image during flight in order to extract the location of potential obstacle, based on the knowledge that texture changes differently for background elements if compared to obstacles. This, however, was demonstrated for walls and large obstacles, but its performance on small objects such as MAVs is not known.

G-1-4 Distance measurement using stereo-vision

A stereo camera provides two images with a *known* translation between the camera frames. This makes it possible to infer 3D properties about the environment. A popular method used for this is dense-stereo (Corke, 2011). Dense stereo operates by matching pixels from one image to the ones of the other image and observing the shift, this is known as *block-matching*. A variant of this method is *semi-global matching* (Hirschmüller, 2005), which also includes neighboring blocks in the disparity estimation. As the transformations between the two figures is already known, it is possible to extract distance. Combined with object recognition, this can be a powerful tool to measure both range *and* bearing to an object.

Stereo vision in cluttered environment can lead to a number of issues (Corke, 2011). One error mode is known as *picket-fence failure*, which happens when similar looking objects/features are present several times at roughly equal distance in an image. It is then impossible to judge when the pixels/features match between two images, since this seems to happen at several instances. In the context of localization, this problem could arise if two or more MAVs were to be observed at roughly equal distances and vertical bearing with respect to the observing MAV, or if sufficiently similar features were observed due to a cluttered environment and/or low image resolution. Another error that may arise is *occlusion failure*: when a point seen by one camera cannot be seen from the other. This is likely not relevant in the context at hand. A third issue known as *broad-peak*, which is found when the matching becomes impossible due to a large general area with no distinctive feature, is also not likely relevant in this context, and is mostly relevant when vision is used for navigation purposes.

G-1-5 Conclusion on efficacy of vision-based methods from literature

The advantage of vision lies in its potential to provide range, bearing, and orientation of an obstacle provided a sufficiently advanced algorithms is used with the right conditions and hardware. The general efficacy, unfortunately, cannot be categorized because it is seen to depend on the integrity, robustness and performance of the algorithm of choice. The identified disadvantages/issues with vision-based localization and target-tracking are the following:

- *Uni-directional view.* When using winged MAVs this issue can be expected to be less dominant, as the main direction of flight is always forward. With multi-rotors, however, it may happen that the camera does not face the forward direction of motion, as the multi-rotor is also capable of equally moving side-ways. It is very probably then that other MAVs may be out of the field of view of the limited field of view of the camera.
- *Obstructed vision.* Vision sensors can be obstructed if measurements are not in line of sight due the presence of other objects.
- *Larger computational demands.* This issue is relevant with respect to the design aims, but negligible if one considers the expected increase in processor speeds in future years.

G-2 Experimental results using vision

Preliminary experiments have been performed to augment the literature findings in H and determine the feasibility of using a mono and/or stereo camera for the purpose of relative MAV detection. The results of the experiments are presented in this section. A portion of the measurements, as in Appendix G-2-1 and Appendix G-2-2, have been performed using the stereo-camera that is available on-board of the drones, thus providing realistic images with the on-board resolution of 128×96 pixels. A second portion, treated in Appendix G-2-3 and Appendix G-2-4, use camera measurements taken with a higher-quality monocular camera. Both these sections then focus on stress-testing nominal vision algorithms for feature recognition and optical flow by reducing the image size so as to understand at what point the algorithms begin to become ineffective. The images in these sections have been taken with the primary camera of a Samsung Galaxy S4 Mini (model name: I9190) phone at a native recording resolution of 640×480 pixels. All algorithms are then tested also at 320×240, 160×120, and 128×96 pixels.

G-2-1 Stereo-Vision

In the stereo-vision test, the available on-board disparity map was tested in order to determine whether the MAV is detectable as a nearby object compared to the background. The experiment was performed at distances of 50cm, 100cm, 150cm, 200cm, 250cm, and 300 cm between the camera and the observed MAV, which was placed (roughly) such that it would appear in the center of the image during all measurements. In all cases the camera and other drone were static (the camera was fixed, and drone hanging from a string to simulate hover), thus giving a best-case scenario with respect to oscillations and noise. It was found that the current on-board stereo-camera's disparity map is insufficient for the detection of a nearby

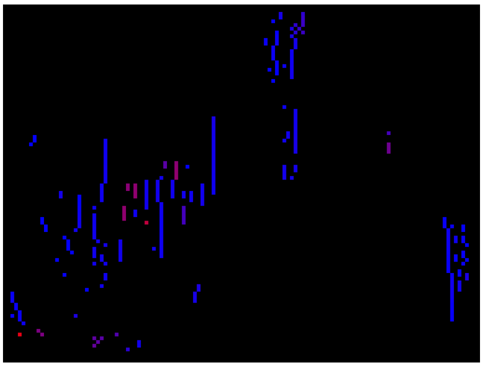


Figure G-1a: Representative image of disparity map at a distance of 0.5 meters from the MAV using on-board camera of other MAV.

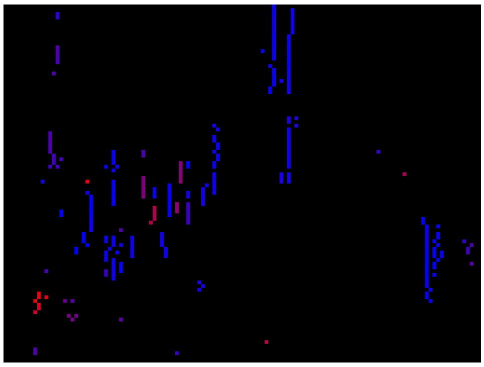


Figure G-1b: Representative 3D disparity map at a distance of 1 meter from the MAV using on-board camera of other MAV.

MAV even at a distance of 50cm, as can be observed in the representative frames shown in Figure G-1a and Figure G-1b.

To validate these results, the native images from the left and right camera were extracted for post-processing using a secondary disparity feature for distances of 50cm, 75cm, 100cm, and 150cm. The MatLab *Computer Vision System Toolbox* [1] was used for this purpose using the `disparity` function, which provides a dense disparity map between two images. Using the `BlockMatching` method, similar to the one available on board of the Ladybird drone, the drone was also not visible. However, it is found that the MAV is visible at distances of 50cm up to 75cm with the `Semi-Global` method. Note that this is only when using a block size of maximum 5 pixels. The MAV is not visible at any distance with higher block sizes. These results for 50cm, 75cm, 100cm, and 150cm are shown in Appendix G-2-1 to Figure G-2d. In the first instance, the MAV is recognizable as the dark red blob approximately in the center of the picture, but fades away significantly as distance increases. Moreover, the picture's low quality means that other dark-red blobs are seen that may be incorrectly interpreted as being an MAV.

G-2-2 Object recognition via feature matching using on-board camera

Using the images extracted only from the left camera as in Appendix G-2-1, object recognition using feature detection and matching was also attempted as a way to recognize the MAV. Representative images are shown in Figure G-3a and Figure G-3b.

Using the SURF feature detection and matching algorithm provided within the MatLab *Computer Vision System Toolbox*, all image features were extracted from an image at 50cm and matched to images at further distances. The results (only including features nearby the rotor, for the sake of clarity) are shown in Figure G-4a to Figure G-4d. The features that are found to match at further distances are not part of the rotor but actually part of the environment behind it (false positives).

[1] More information available at `http://mathworks.com/help/vision/index.html`

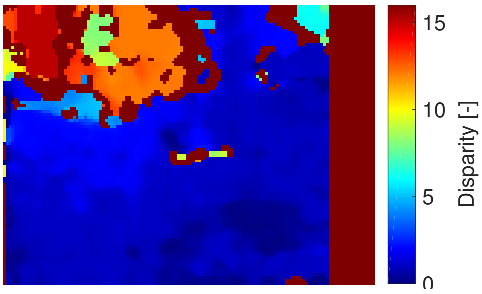


Figure G-2a: Representative image of dense disparity map using Semi-Global block-matching at a distance of 50cm from the MAV using on-board camera of other MAV.

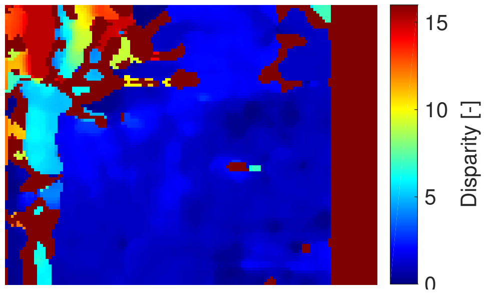


Figure G-2b: Representative image of dense disparity map using Semi-Global block-matching at a distance of 75cm from the MAV using on-board camera of other MAV.

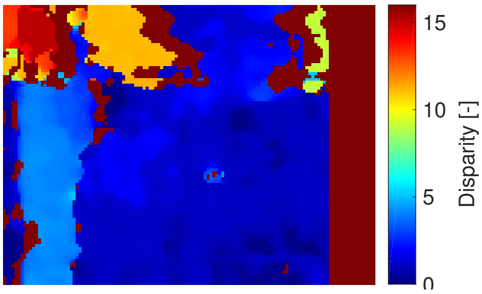


Figure G-2c: Representative image of dense disparity map using Semi-Global block-matching at a distance of 100cm from the MAV using on-board camera of other MAV.

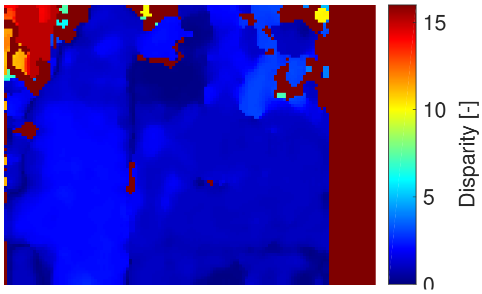


Figure G-2d: Representative image of dense disparity map using Semi-Global block-matching at a distance of 150cm from the MAV using on-board camera of other MAV.

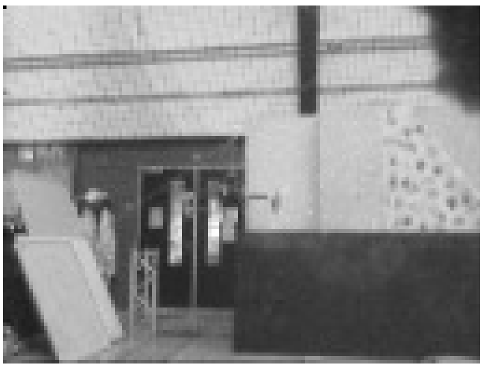


Figure G-3a: Representative image at a distance of 0.5 meter from the MAV using on-board camera of another MAV. Image resolution is 128x96 pixels.

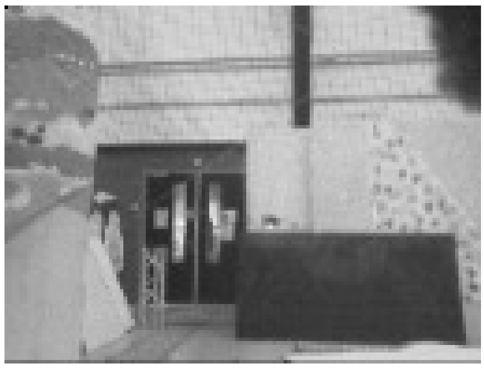


Figure G-3b: Representative image at a distance of 1 meter from the MAV using on-board camera of another MAV. Image resolution is 128x96 pixels.

G-2-3 Object recognition via feature matching (resolution stress-testing)

To test the impact of image resolution on the ability of object recognition, the experiments from Appendix G-2-2 were performed using a higher resolution camera with a native resolution of 640×480 pixels. The process of feature recognition at the native resolution, where one

Figure G-4a: Feature detection using representative figure taken at distance of 50 cm (using SURF)

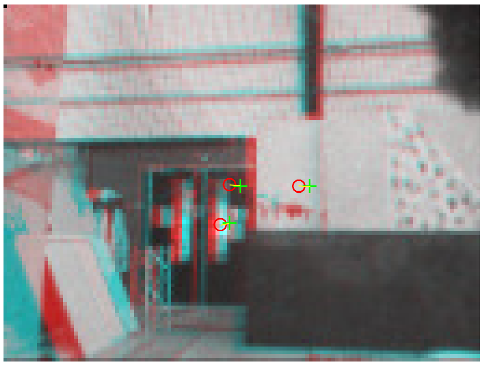


Figure G-4b: Feature matching of drone-related features from extracted as in Figure G-4a to representative picture taken at 75 cm

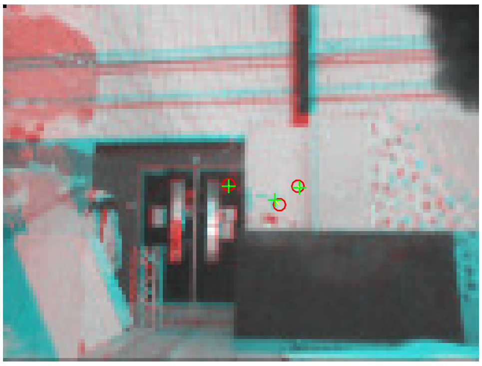


Figure G-4c: Feature matching of drone-related features from extracted as in Figure G-4a to representative picture taken at 100 cm

Figure G-4d: Feature matching of drone-related features from extracted as in Figure G-4a to representative picture taken at 150 cm

correct feature was matched between distances of 50cm and 100cm. None were correctly matched at further distances. These results are shown in Figure G-5a to G-5c. At lower resolutions, 320×240, or even 128×96, since less features related to the MAV can be detected in the first place, no features are successfully matched. The results are shown in Figure G-6a to G-6c for 320×240 and in Figure G-7a to G-7c for 128×96.

G-2-4 Optical flow stress-testing

This section shows the analysis of sample videos in order to assess the ability of optical flow to detect other MAVs. Native videos are taken at 640×480 pixels, and subsequently scaled to resolutions of 320×240, 160×120, and 128×96 (where the lowest resolution is representative of the resolution by the on-board camera). The optical flow implementation and object recognition method used was Lukas-Kanade as provided within Simulink by the MatLab *Computer Vision System Toolbox*. Objects were recognized as regions where pixels moved at a velocity higher than the average velocity over the whole image.

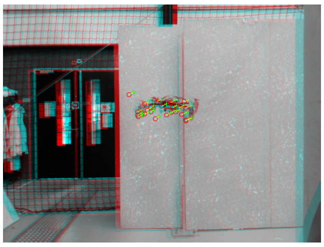


Figure G-5a: Feature detection and matching for two images, both taken at a distance of 50 cm, at 640×480 resolution.

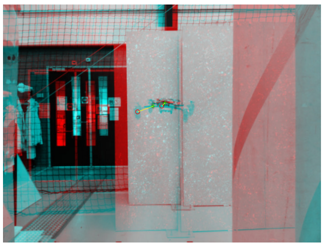


Figure G-5b: Feature detection and matching for two images, one taken at a distance of 50 cm and another taken at a distance of 100 cm, at 640×480 resolution.

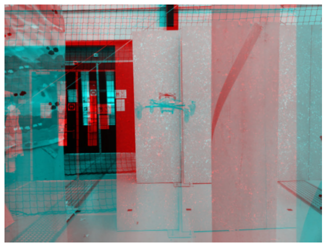


Figure G-5c: Feature detection and matching for two images, one taken at a distance of 50 cm and another taken at a distance of 150 cm, at 640×480 resolution.

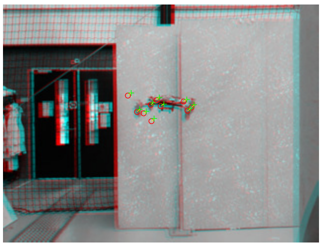


Figure G-6a: Feature detection and matching for two images, both taken at a distance of 50 cm, at 320×240 resolution.

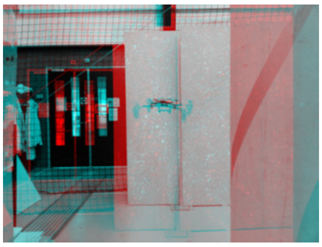


Figure G-6b: Feature detection and matching for two images, one taken at a distance of 50 cm and another taken at a distance of 100 cm, at 320×240 resolution.

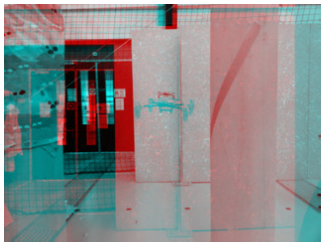


Figure G-6c: Feature detection and matching for two images, one taken at a distance of 50 cm and another taken at a distance of 150 cm, at 320×240 resolution.

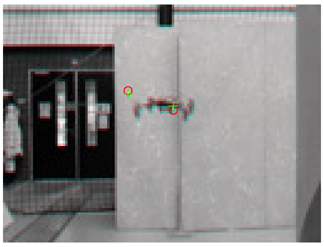


Figure G-7a: Feature detection and matching for two images, both taken at a distance of 50 cm, at 128×96 resolution.

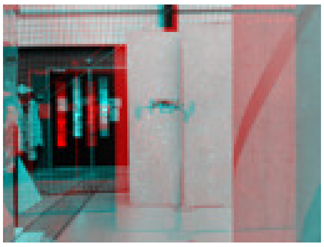


Figure G-7b: Feature detection and matching for two images, one taken at a distance of 50 cm and another taken at a distance of 100 cm, at 128×96 resolution.

Figure G-7c: Feature detection and matching for two images, one taken at a distance of 50 cm and another taken at a distance of 150 cm, at 128×96 resolution.

Moving observer and static MAV

In the first case, a near-collision footage was recorded for the case of a static MAV and an approaching observer. A representative snapshot of the results for native-resolution footage is shown in Figure G-8a to G-8c. An object can be recognized at lower distances but consistent flickering is observed as opposed to better defined areas such as the door. The same has been tried at a resolution of 160×120, where the MAV is no longer distinguishable at any point. Representative results are shown in Figure G-9a to G-9c.

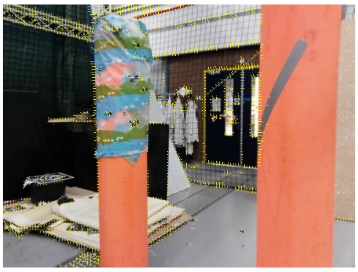


Figure G-8a: Image of optical flow during oncoming collision at native resolution with static MAV obstacle and moving observer.

Figure G-8b: Image reconstruction from flow at optical flow during oncoming collision at native resolution with static MAV obstacle and moving observer.

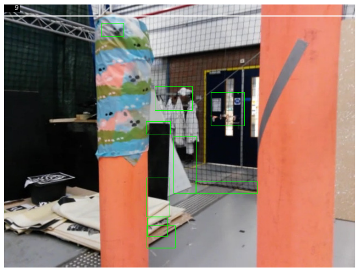


Figure G-8c: Object identification flow grouping during oncoming collision at native resolution with static MAV obstacle and moving observer (objects enclosed by green boxes)

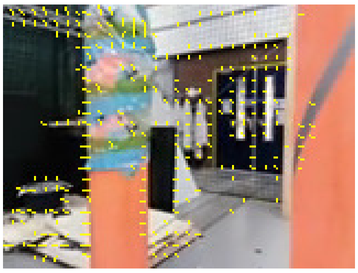


Figure G-9a: Image of optical flow during oncoming collision at 160×120 resolution with static MAV obstacle and moving observer.

Figure G-9b: Image reconstruction from flow at optical flow during oncoming collision at 160×120 resolution with static MAV obstacle and moving observer.

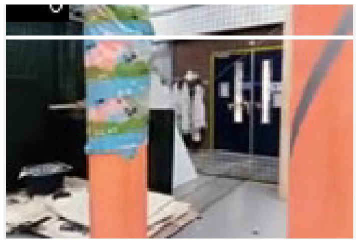


Figure G-9c: Object identification flow grouping during oncoming collision at 160×120 resolution with static MAV obstacle and moving observer (objects enclosed by green boxes)

Moving MAV and static observer

For a moving MAV with static observer, one can expect the MAV to be easier to recognize using the optical-flow/object recognition algorithm at hand, as there should be less observed movement in the background. A representative snapshot of the results is shown in Figure G-10a to G-10c. Although the MAV is recognized, the main issue to be noted are the large amount of false positives due other objects in the cluttered environment which seem to be moving due to camera oscillations (which are to be expected if the camera is attached to a hovering drone). At lower resolutions, such as 160×120, the MAV is no longer reliably recognizable as shown in Figure G-11a to G-11c.

Moving MAV and observer

A third set of footage was collected for a collision path with both a moving MAV and observer. These results allowed tracking of the MAV at higher resolution, but the results worsened at

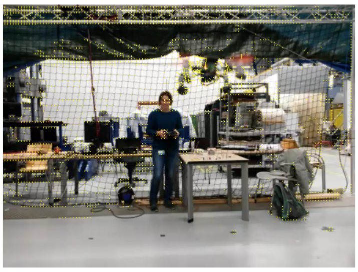


Figure G-10a: Image of optical flow during oncoming collision at native resolution with static observer and moving MAV.

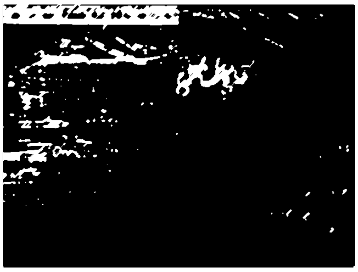


Figure G-10b: Image reconstruction from flow at optical flow during oncoming collision at native resolution with static observer and moving MAV.

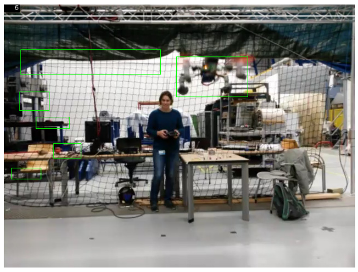


Figure G-10c: Object identification flow grouping during oncoming collision at native resolution with static observer and moving MAV (objects enclosed by green boxes)

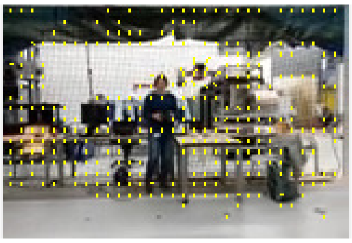


Figure G-11a: Image of optical flow during oncoming collision at 160×120 resolution with static MAV obstacle and moving observer.

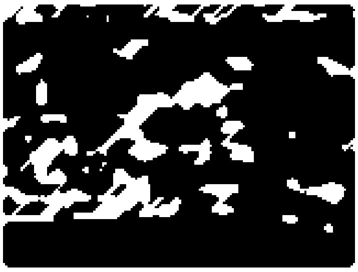


Figure G-11b: Image reconstruction from flow at optical flow during oncoming collision at 160×120 resolution with static MAV obstacle and moving observer.

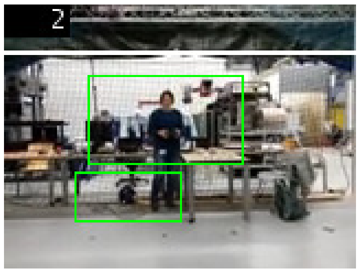


Figure G-11c: Object identification flow grouping during oncoming collision at 160×120 resolution with static MAV obstacle and moving observer (objects enclosed by green boxes)

lower resolutions. For this case, an alternative approach was attempted where all pixels moving at a slower rate than average were identified as the object. This is because two moving observers at similar velocities results in the MAV appearing static within the frame. This opposite algorithm, however, was also not suitable due to the oscillations of the camera and did not work successfully.

G-2-5 The influence of field of view

Even excluding all problems related to the fail-safe detection of another MAV, the camera's limitations with respect to field of view can pose significant issues for collision avoidance (Guzzi, Giusti, Gambardella, Theraulaz, et al., 2013). One example of a failure condition is exemplified in Figure G-14a and Figure G-14b. These instances show a collision path whereby the camera's field of view is insufficient to detect the incoming MAV until a collision is imminent. In this case, within the span of 5 frames (which, for a video recorded at 30fps, means ≈ 0.16s), the MAV goes from entering the frame to a near-collision.

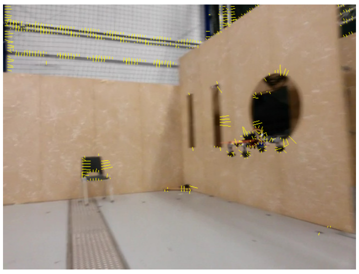


Figure G-12a: Image of optical flow during oncoming collision at native resolution with static observer and moving MAV.

Figure G-12b: Image reconstruction from flow at optical flow during oncoming collision at native resolution with static observer and moving MAV.

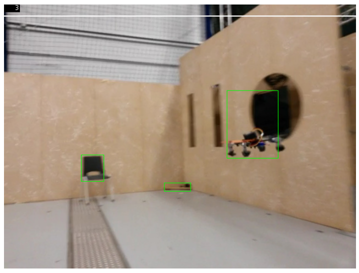


Figure G-12c: Object identification flow grouping during oncoming collision at native resolution with static observer and moving MAV (objects enclosed by green boxes)

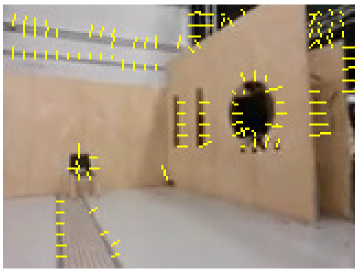


Figure G-13a: Image optical flow during oncoming collision at 160×120 resolution with both moving MAV obstacle and moving observer at $\approx 90\circ$ collision path

Figure G-13b: Image reconstruction from flow at optical flow during oncoming collision at 160×120 resolution with static MAV obstacle and moving observer.

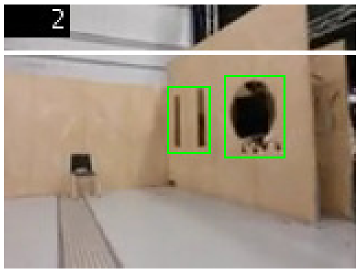


Figure G-13c: Object identification flow grouping during oncoming collision at 160×120 resolution with static MAV obstacle and moving observer (objects enclosed by green boxes)

Figure G-14a: Still from frame 18 of recorded video (at 30fps). The MAV is judged recognizable starting from this frame.

Figure G-14b: Still from frame 23 of recorded video (at 30fps). The MAV is at its closest point (near-collision).

Appendix H

ROS Implementation Details

This section briefly explains additional details regarding the system implementation in the ROS/Gazebo simulation environment. Figure H-1 shows a graph of the communication between the active nodes during a simulation with 2 MAVs. This is analogous to the situation with 3 (or more) MAVs, in which case more namespaces are added (uav3, uav4, etc.). Currently, the simulation environment is set-up to be launched with up to 3 MAVs.

The node developed within this report is called "mav_ctrl", shown in **uav1** and **uav2** as **/uav1/mav_ctrl_1** and **/uav1/mav_ctrl_2**. This node handles the following:

- Simulation of Bluetooth antenna and sensor receivers.
- Code EKF and avoidance functions (built by including code directly from Paparazzi UAS).
- Output of velocity commands to relevant simulated MAV.
- Logging of on-board data to a `txt` file for analysis.
- Check collision to other MAVs and kill the simulation if a collision occurs. A collision is any time that the distance between the two drones is larger than twice their radii.

The node reads a set of parameters that may be set at run time:

- *Collision avoidance active or not active.* This is set by instructing parameter "avoidance" 1 or 0 for active and not active, respectively.
- *Diameter of the MAVs.* The is parameter name "mavsize".
- *Arena Side Length.* This is set by parameter "arenasize".
- *Name of the MAV.* This is the ID of the MAV at runtime, it may be a string.

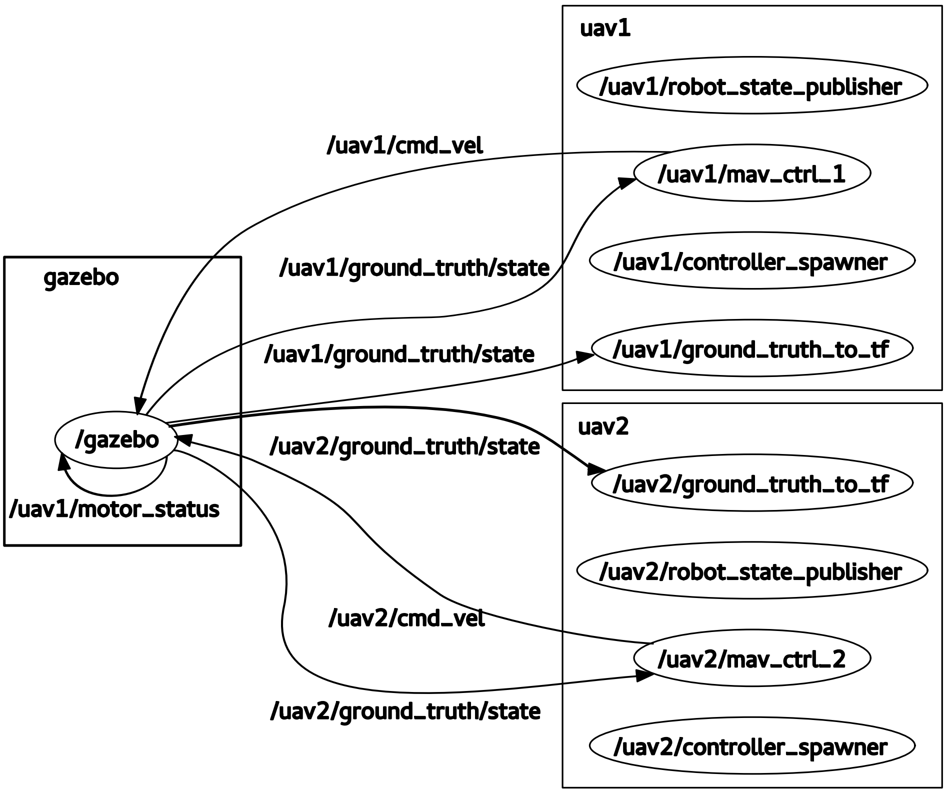


Figure H-1: Graph of communication between ROS nodes for a simulation with 2 MAVs. **Gazebo**, **uav1**, and **uav2** are the active namespaces. Gazebo published ground truth data ("uav1/ground_truth/state") to **uav1** and **uav2**. The controller of this report is coded in the nodes **/uav1/mav_ctrl_1** and **/uav1/mav_ctrl_2** for **uav1** and **uav2**, respectively. These send velocity commands back to the gazebo simulations.

Bibliography

Abeywardena, D., Kodagoda, S., Dissanayake, G., & Munasinghe, R. (2013, Dec). Improved state estimation in quadrotor mavs: A novel drift-free velocity estimator. *Robotics Automation Magazine, IEEE*, *20*(4), 32-39.

Afzal, M. H., Renaudin, V., & Lachapelle, G. (2011). Magnetic field based heading estimation for pedestrian navigation environments. In *Indoor positioning and indoor navigation (ipin), 2011 international conference on* (pp. 1–10).

Antonelli, G., Arrichiello, F., Chiaverini, S., & Sukhatme, G. S. (2010). Observability analysis of relative localization for auvs based on ranging and depth measurements. In *Robotics and automation (icra), 2010 ieee international conference on* (pp. 4276–4281).

Araujo, D., Davids, K., & Hristovski, R. (2006). The ecological dynamics of decision making in sport. *Psychology of Sport and Exercise*, *7*(6), 653–676.

Assa, A., & Janabi-Sharifi, F. (2015). A kalman filter-based framework for enhanced sensor fusion. *Sensors Journal, IEEE*, *15*(6), 3281-3292.

Azzam, A., & Wang, X. (2010). Quad rotor arial robot dynamic modeling and configuration stabilization. In *Informatics in control, automation and robotics (car), 2010 2nd international asia conference on* (Vol. 1, pp. 438–444).

Basiri, M. (2015). Audio-based positioning and target localization for swarms of micro aerial vehicles.

Basiri, M., Schill, F., Floreano, D., & Lima, P. U. (2014). Audio-based localization for swarms of micro air vehicles. In *Robotics and automation (icra), 2014 ieee international conference on* (pp. 4729–4734).

Baunach, M., Mühlberger, C., Appold, C., Schröder, M., & Füller, F. (2009). Analysis of radio signal parameters for calibrating rssi localization systems. *Institut für Informatik, Universität Würzburg, Tech. Rep*, *455*.

Bay, H., Ess, A., Tuytelaars, T., & Van Gool, L. (2008). Speeded-up robust features (surf). *Computer vision and image understanding*, *110*(3), 346–359.

Beard, R. W. (2007). State estimation for micro air vehicles. In *Innovations in intelligent machines-1* (pp. 173–199). Springer.

Benjamin, B., Erinc, G., & Carpin, S. (2015). Real-time wifi localization of heterogeneous robot teams using an online random forest. *Autonomous Robots*, 1–13.

Bouabdallah, S., Murrieri, P., & Siegwart, R. (2004). Design and control of an indoor micro quadrotor. In *Robotics and automation, 2004. proceedings. icra'04. 2004 ieee international conference on* (Vol. 5, pp. 4393–4398).

Bry, A., Bachrach, A., & Roy, N. (2012). State estimation for aggressive flight in gps-denied environments using onboard sensing. In *Robotics and automation (icra), 2012 ieee international conference on* (pp. 1–8).

Caron, C., Chamberland-Tremblay, D., Lapierre, C., Hadaya, P., Roche, S., & Saada, M. (2008). Indoor positioning. In *Encyclopedia of gis* (pp. 553–559). Springer.

Chi Mak, L., Whitty, M., & Furukawa, T. (2008). A localisation system for an indoor rotary-wing mav using blade mounted leds. *Sensor Review*, *28*(2), 125–131.

Conroy, P., Bareiss, D., Beall, M., & van den Berg, J. (2014). 3-d reciprocal collision avoidance on physical quadrotor helicopters with on-board sensing for relative positioning. *arXiv preprint arXiv:1411.3794*.

Corke, P. (2011). *Robotics, vision and control: fundamental algorithms in matlab* (Vol. 73). Springer Science & Business Media.

Cornejo, A., & Nagpal, R. (2015). Distributed range-based relative localization of robot swarms. In *Algorithmic foundations of robotics xi* (pp. 91–107). Springer.

Crespi, V., Galstyan, A., & Lerman, K. (2008). Top-down vs bottom-up methodologies in multi-agent system design. *Autonomous Robots*, *24*(3), 303–313.

Dahlgren, E., & Mahmood, H. (2014). *Evaluation of indoor positioning based on bluetoothr smart technology* (Unpublished doctoral dissertation). Chalmers University of Technology.

De Croon, G., De Wagter, C., Remes, B., & Ruijsink, R. (2012). Sub-sampling: Real-time vision for micro air vehicles. *Robotics and Autonomous Systems*, *60*(2), 167–181.

De Croon, G., De Weerdt, E., De Wagter, C., Remes, B., & Ruijsink, R. (2012). The appearance variation cue for obstacle avoidance. *Robotics, IEEE Transactions on*, *28*(2), 529–534.

De Croon, G., Ho, H., De Wagter, C., Van Kampen, E., Remes, B., & Chu, Q. (2013). Optic-flow based slope estimation for autonomous landing. *International Journal of Micro Air Vehicles*, *5*(4), 287–298.

Fajen, B. R., & Warren, W. H. (2003). Behavioral dynamics of steering, obstable avoidance, and route selection. *Journal of Experimental Psychology: Human Perception and Performance*, *29*(2), 343.

Fiorini, P., & Shiller, Z. (1998). Motion planning in dynamic environments using velocity obstacles. *The International Journal of Robotics Research*, *17*(7), 760–772.

Fischer, G., Dietrich, B., & Winkler, F. (2004). Bluetooth indoor localization system. In *Proceedings of the 1st workshop on positioning, navigation and communication.*

Gebre-Egziabher, D., & Elkaim, G. H. (2008). Mav attitude determination by vector matching. *Aerospace and Electronic Systems, IEEE Transactions on*, *44*(3), 1012–1028.

Goldsmith, A. (2005). *Wireless communications.* Cambridge university press.

Gremillion, G., & Humbert, J. S. (2010). System identification of a quadrotor micro air vehicle. In *Aiaa conference on atmospheric flight mechanics.*

Gugerty, L. J. (1997). Situation awareness during driving: Explicit and implicit knowledge in dynamic spatial memory. *Journal of Experimental Psychology: Applied*, *3*(1), 42.

Guzzi, J., Giusti, A., Gambardella, L. M., Di Caro, G., et al. (2013). Local reactive robot navigation: A comparison between reciprocal velocity obstacle variants and human-like behavior. In *Intelligent robots and systems (iros), 2013 ieee/rsj international conference on* (pp. 2622–2629).

Guzzi, J., Giusti, A., Gambardella, L. M., & Di Caro, G. A. (2014). Bioinspired obstacle avoidance algorithms for robot swarms. In *Bio-inspired models of network, information, and computing systems* (pp. 120–134). Springer.

Guzzi, J., Giusti, A., Gambardella, L. M., Theraulaz, G., Di Caro, G., et al. (2013). Human-friendly robot navigation in dynamic environments. In *Robotics and automation (icra), 2013 ieee international conference on* (pp. 423–430).

Hackney, A. L., & Cinelli, M. E. (2013). Action strategies used by children to avoid two vertical obstacles in non-confined space. *Experimental brain research*, *229*(1), 13–22.

Hauert, S., Leven, S., Zufferey, J.-C., & Floreano, D. (2010). Communication-based swarming for flying robots. In *Proceedings of the workshop on network science and systems issues in multi-robot autonomy, ieee international conference on robotics and automation.*

Hennes, D., Claes, D., Meeussen, W., & Tuyls, K. (2012). Multi-robot collision avoidance with localization uncertainty. In *Proceedings of the 11th international conference on autonomous agents and multiagent systems-volume 1* (pp. 147–154).

Hirschmüller, H. (2005). Accurate and efficient stereo processing by semi-global matching and mutual information. In *Computer vision and pattern recognition, 2005. cvpr 2005. ieee computer society conference on* (Vol. 2, pp. 807–814).

Ho, H., De Wagter, C., Remes, B., & de Croon, G. (2015). Optical-flow based self-supervised learning of obstacle appearance applied to mav landing. *arXiv preprint arXiv:1509.01423*.

Howard, A., Matarić, M. J., & Sukhatme, G. S. (2003). Putting the'i'in'team': an ego-centric approach to cooperative localization. In *Robotics and automation, 2003. proceedings. icra'03. ieee international conference on* (Vol. 1, pp. 868–874).

Iyer, A., Rayas, L., & Bennett, A. (2013). Formation control for cooperative localization of mav swarms. In *Proceedings of the 2013 international conference on autonomous agents and multi-agent systems* (pp. 1371–1372).

Kendall, A. G., Salvapantula, N. N., Stol, K., et al. (2014). On-board object tracking control of a quadcopter with monocular vision. In *Unmanned aircraft systems (icuas), 2014 international conference on* (pp. 404–411).

Kendoul, F., Fantoni, I., & Nonami, K. (2009). Optic flow-based vision system for autonomous 3d localization and control of small aerial vehicles. *Robotics and Autonomous Systems*, *57*(6), 591–602.

Kendoul, F., Nonami, K., Fantoni, I., & Lozano, R. (2009). An adaptive vision-based autopilot for mini flying machines guidance, navigation and control. *Autonomous Robots*, *27*(3), 165–188.

Kushki, A., Plataniotis, K., & Venetsanopoulos, A. (2008). Indoor positioning with wireless local area networks (wlan). In *Encyclopedia of gis* (pp. 566–571). Springer.

Lee, D., Franchi, A., Son, H. I., Ha, C., Bulthoff, H. H., & Giordano, P. R. (2013). Semiautonomous haptic teleoperation control architecture of multiple unmanned aerial vehicles. *Mechatronics, IEEE/ASME Transactions on*, *18*(4), 1334–1345.

Lee, K. U., Yun, Y. H., Chang, W., Park, J. B., & Choi, Y. H. (2011). Modeling and altitude control of quad-rotor uav. In *Control, automation and systems (iccas), 2011 11th international conference on* (pp. 1897–1902).

Leishman, R. C., Macdonald, J. C., Beard, R. W., & McLain, T. W. (2014). Quadrotors and accelerometers: State estimation with an improved dynamic model. *Control Systems,* *IEEE*, *34*(1), 28–41.

Lima, P. U., Santos, P., Oliveira, R., Ahmad, A., & Santos, J. (2011). Cooperative localization based on visually shared objects. In *Robocup 2010: Robot soccer world cup xiv* (pp. 350–361). Springer.

Lowe, D. G. (1987). Three-dimensional object recognition from single two-dimensional images. *Artificial intelligence*, *31*(3), 355–395.

Lowe, D. G. (1999). Object recognition from local scale-invariant features. In *Computer vision, 1999. the proceedings of the seventh ieee international conference on* (Vol. 2, pp. 1150–1157).

Macdonald, J., Leishman, R., Beard, R., & McLain, T. (2014). Analysis of an improved imu-based observer for multirotor helicopters. *Journal of Intelligent & Robotic Systems*, *74*(3-4), 1049–1061.

Mahony, R., Kumar, V., & Corke, P. (2012). Multirotor aerial vehicles: Modeling, estimation, and control of quadrotor. *IEEE Robotics & amp amp Automation Magazine*(19), 20–32.

Malyavej, V., Kumkeaw, W., & Aorpimai, M. (2013). Indoor robot localization by rssi/imu sensor fusion. In *Electrical engineering/electronics, computer, telecommunications and information technology (ecti-con), 2013 10th international conference on* (pp. 1–6).

Mariottini, G. L., Morbidi, F., Prattichizzo, D., Valk, N. V., Michael, N., Pappas, G., & Daniilidis, K. (2009). Vision-based localization for leader–follower formation control. *Robotics, IEEE Transactions on*, *25*(6), 1431–1438.

Martinelli, A. (2011). Closed-form solution for attitude and speed determination by fusing monocular vision and inertial sensor measurements. In *Robotics and automation (icra), 2011 ieee international conference on* (pp. 4538–4545).

Martinelli, A., Pont, F., & Siegwart, R. (2005). Multi-robot localization using relative observations. In *Robotics and automation, 2005. icra 2005. proceedings of the 2005 ieee international conference on* (pp. 2797–2802).

Martinelli, A., & Siegwart, R. (2005). Observability analysis for mobile robot localization. In *Intelligent robots and systems, 2005.(iros 2005). 2005 ieee/rsj international conference on* (pp. 1471–1476).

Michael, N., Mellinger, D., Lindsey, Q., & Kumar, V. (2010). The grasp multiple micro-uav testbed. *Robotics & Automation Magazine, IEEE*, *17*(3), 56–65.

Mulder, J., van Staveren, W., van der Vaart, J., & others. (2013). *Flight dynamics lecture notes.* Delft University of Technology. Online Reader.

Nguyen, K., & Luo, Z. (2013). Evaluation of bluetooth properties for indoor localisation. In *Progress in location-based services* (pp. 127–149). Springer.

No, H., Cho, A., & Kee, C. (2015). Attitude estimation method for small uav under accelerative environment. *GPS Solutions*, *19*(3), 343–355.

Olfati-Saber, R. (2009). Kalman-consensus filter: Optimality, stability, and performance. In *Decision and control, 2009 held jointly with the 2009 28th chinese control conference. cdc/ccc 2009. proceedings of the 48th ieee conference on* (pp. 7036–7042).

Pascoal, A., Kaminer, I., & Oliveira, P. (2000). Navigation system design using time-varying complementary filters. *Aerospace and Electronic Systems, IEEE Transactions on*, *36*(4), 1099–1114.

Pathirana, P. N., Ekanayake, S. W., & Savkin, A. V. (2011). Fusion based 3d tracking of mobile transmitters via robust set-valued state estimation with rss measurements. *Communications Letters, IEEE*, *15*(5), 554–556.

Perkins, C., Lei, L., Kuhlman, M., Lee, T., Gateau, G., Bergbreiter, S., & Abshire, P. (2011). Distance sensing for mini-robots: Rssi vs. tdoa. In *Circuits and systems (iscas), 2011 ieee international symposium on* (pp. 1984–1987).

Phang, S. K., Cai, C., Chen, B. M., & Lee, T. H. (2012). Design and mathematical modeling of a 4-standard-propeller (4sp) quadrotor. In *Intelligent control and automation (wcica), 2012 10th world congress on* (pp. 3270–3275).

Pounds, P., Mahony, R., & Corke, P. (2006). Modelling and control of a quad-rotor robot. In *Proceedings australasian conference on robotics and automation 2006.*

Powers, C., Mellinger, D., Kushleyev, A., Kothmann, B., & Kumar, V. (2013). Influence of aerodynamics and proximity effects in quadrotor flight. In *Experimental robotics* (pp. 289–302).

Raju, M., Oliveira, T., & Agrawal, D. P. (2012). A practical distance estimator through distributed rssi/lqi processingan experimental study. In *Communications (icc), 2012 ieee international conference on* (pp. 6575–6579).

Rappaport, T. S., et al. (1996). *Wireless communications: principles and practice* (Vol. 2). prentice hall PTR New Jersey.

Remes, B., Esden-Tempski, P., Van Tienen, F., Smeur, E., De Wagter, C., & De Croon, G. (2014). Lisa-s 2.8 g autopilot for gps-based flight of mavs. In *Imav 2014: International micro air vehicle conference and competition 2014, delft, the netherlands, august 12-15, 2014.*

Ren, Z., Wang, G., Chen, Q., & Li, H. (2011). Modelling and simulation of rayleigh fading, path loss, and shadowing fading for wireless mobile networks. *Simulation Modelling Practice and Theory*, *19*(2), 626–637.

Roberts, J. F., Stirling, T., Zufferey, J.-C., & Floreano, D. (2012). 3-d relative positioning sensor for indoor flying robots. *Autonomous Robots*, *33*(1-2), 5–20.

Rodas, J., Escudero, C. J., Iglesia, D., et al. (2008). Bayesian filtering for a bluetooth positioning system. In *Wireless communication systems. 2008. iswcs'08. ieee international symposium on* (pp. 618–622).

Roumeliotis, S. I., & Rekleitis, I. M. (2004). Propagation of uncertainty in cooperative multirobot localization: Analysis and experimental results. *Autonomous Robots*, *17*(1), 41–54.

Sabatini, A. M., & Genovese, V. (2013). A stochastic approach to noise modeling for barometric altimeters. *Sensors*, *13*(11), 15692–15707.

Sasiadek, J., & Hartana, P. (2000). Sensor data fusion using kalman filter. In *Information fusion, 2000. fusion 2000. proceedings of the third international conference on* (Vol. 2, pp. WED5–19).

Sasiadek, J. Z. (2002). Sensor fusion. *Annual Reviews in Control*, *26*(2), 203–228.

Saska, M. (2015). Mav-swarms: Unmanned aerial vehicles stabilized along a given path using onboard relative localization. In *Unmanned aircraft systems (icuas), 2015 international conference on* (pp. 894–903).

Scaramuzza, D., Achtelik, M. C., Doitsidis, L., Friedrich, F., Kosmatopoulos, E., Martinelli, A., ... others (2014). Vision-controlled micro flying robots: from system design to autonomous navigation and mapping in gps-denied environments. *Robotics & Automation Magazine, IEEE*, *21*(3), 26–40.

Schiff, W., & Detwiler, M. L. (1979). Information used in judging impending collision. *Perception*, *8*(6), 647–658.

Seybold, J. S. (2005). *Introduction to rf propagation.* John Wiley & Sons.

Sharma, R., Beard, R. W., Taylor, C. N., & Quebe, S. (2012). Graph-based observability analysis of bearing-only cooperative localization. *Robotics, IEEE Transactions on*, *28*(2), 522–529.

Sharma, R., & Taylor, C. (2008). Cooperative navigation of mavs in gps denied areas. In *Multisensor fusion and integration for intelligent systems, 2008. mfi 2008. ieee international conference on* (pp. 481–486).

Sharp, I., & Yu, K. (2014). Sensor-based dead-reckoning for indoor positioning. *Physical Communication*, *13*, 4–16.

Shilov, K. (2014). The next generation design of autonomous mav flight control system smartap. In *Imav 2014: International micro air vehicle conference and competition 2014, delft, the netherlands, august 12-15, 2014.*

Snape, J., van den Berg, J., Guy, S. J., & Manocha, D. (2009). Independent navigation of multiple mobile robots with hybrid reciprocal velocity obstacles. In *Intelligent robots and systems, 2009. iros 2009. ieee/rsj international conference on* (pp. 5917–5922).

Snape, J., van den Berg, J., Guy, S. J., & Manocha, D. (2011). The hybrid reciprocal velocity obstacle. *Robotics, IEEE Transactions on*, *27*(4), 696–706.

Spears, W. M., Hamann, J. C., Maxim, P. M., Kunkel, T., Heil, R., Zarzhitsky, D., ... Karlsson, C. (2007). Where are you? In *Swarm robotics* (pp. 129–143). Springer.

Subhan, F., Hasbullah, H., & Ashraf, K. (2013). Kalman filter-based hybrid indoor position estimation technique in bluetooth networks. *International Journal of Navigation and Observation*, *2013*.

Svečko, J., Malajner, M., & Gleich, D. (2015). Distance estimation using rssi and particle filter. *ISA transactions*, *55*, 275–285.

Szabo, T. (2015). *Autonomous collision avoidance for swarms of mavs based solely on rssi measurements* (Unpublished master's thesis). Delft University of Technology.

Trawny, N., Zhou, X. S., & Roumeliotis, S. I. (2009). 3d relative pose estimation from six distances. In *Robotics: Science and systems.*

Troiani, C., Martinelli, A., Laugier, C., & Scaramuzza, D. (2013). 1-point-based monocular motion estimation for computationally-limited micro aerial vehicles. In *Mobile robots (ecmr), 2013 european conference on* (pp. 13–18).

Van Den Berg, J., Guy, S. J., Lin, M., & Manocha, D. (2011). Reciprocal n-body collision avoidance. In *Robotics research* (pp. 3–19). Springer.

Vossiek, M., Wiebking, L., Gulden, P., Weighardt, J., & Hoffmann, C. (2003). Wireless local positioning-concepts, solutions, applications. In *Radio and wireless conference, 2003. rawcon'03. proceedings* (pp. 219–224).

Xu, J., Ma, M., & Law, C. L. (2015). Cooperative angle-of-arrival position localization. *Measurement*, *59*, 302–313.

Youssef, M. (2008). Indoor localization. In *Encyclopedia of gis* (pp. 547–552). Springer.

Zhou, X. S., Roumeliotis, S., et al. (2008). Robot-to-robot relative pose estimation from range measurements. *Robotics, IEEE Transactions on*, *24*(6), 1379–1393.

Zhou, X. S., Roumeliotis, S., et al. (2013). Determining 3-d relative transformations for any combination of range and bearing measurements. *Robotics, IEEE Transactions on*, *29*(2), 458–474.