2019 IEEE International Symposium on Safety, Security, and Rescue Robotics (SSRR)
Würzburg, Germany, September 2-4, 2019

A concept for catching drones with a net carried by cooperative UAVs

Julian Rothe[1], Michael Strohmeier[2] and Sergio Montenegro[3]

***Abstract*— Unmanned Ariel Vehicles (UAVs) are increasingly used in research and by the public. Though this technology has many advantages and will possibly change industries, it also brings new challenges. Since UAVs are not bound to use existing infrastructure, current laws and rules can not be applied to them. Even worse, they can easily overcome existing infrastructural barriers and be a potential harm to society. Therefore, the German federal government funded several projects, which focus on developing a Drone Protection System (DPS). In this paper we will show a concept for an UAV-System capable of catching possibly dangerous UAVs in mid air as part of one of these projects called *MIDRAS*. After describing the project briefly and showing the state of the art with different existing solutions to stop none cooperative drones, we present the concept of developing our system in detail. Here, we first describe the hard- and software platform before showing our solution for high accuracy outdoor relative positioning. Next, we derive a way to get the optimal point of intersecting the target and further describe our planned master-slave system for the formation flight. We conclude the work with first preliminary results on catching drones in nets and flying our UAVs in formation as well as a short outlook on the next steps.**

I. INTRODUCTION

Just within a few years Unmanned Aerial Vehicle (UAV)s have become one of the most accelerating technologies, both in research and the public. In Germany alone, there are by the time of this publication almost 500.000 drones and this number will increase to over 900.000 in 2030 [1]. Although the commercial use of drones is on the rise, the largest amount of these drones are bought and used privately, mostly for filming and entertainment purposes. Since UAVs are not bound to the ground, there is a new third dimension added to our current traffic systems. Therefore, we are suddenly faced with several new challenges:

- Existing rules for traffic management cannot be used for the new systems, since drones are not constrained to follow given roads
- Existing barriers for controlling the access to facilities and land are not meant to protect illegal access from air
- Existing laws can either not be applied to the new technology or are meant for bigger aircrafts carrying human passengers
- Existing methods to survey and control vehicles on the ground can not be used for UAVs

Consequently, the German federal government responded to these challenges with several different research projects focusing on UAVs. As part of the program for security research [2] the project called Micro Drone Protection System (MIDRAS) [3] focuses on the protection against unauthorized usage of small drones ($<$ 5 kg) in different scenarios. The aim of the research project is to develop a DPS which is able to detect, classify and categorize as well as to produce appropriate counter measures against uncooperative UAVs. The system is developed considering a wide range of different scenarios from the protection of VIPs to the surveillance of big infrastructures like airports or stadiums. To reach this goal, the project can be split into two subcategories:

- **Detection and classification:** The first step is to build a multi-sensor detection system which is able to detect flying objects in a great distance around a perimeter to protect. For this purpose, the data of audio, optical and radar sensors are combined in a multi state sensor fusion algorithm to derive the position, velocity and predicted trajectory of the object. The sensor data is also used to get a classification of the object, which is used to guide the user to appropriate counter measures if these are needed.
- **Counter measures:** If an object is detected and classified as possibly harmful there are different types of counter measures an operator can choose from. One of these counter measures will be a UAV system using formation flight to carry a net to catch drones mid air which is developed at the *University of Wuerzburg* and will be further described in this paper.

Fig. 1. Model of the UAV system carrying a net

[1]Julian Rothe is with the Chair of Aerospace Information Technology, Mathematics and Computer Science, University of Wuerzburg, Germany julian.rothe@uni-wuerzburg.de
[2]Michael Strohmeier is with the Chair of Aerospace Information Technology, Mathematics and Computer Science, University of Wuerzburg, Germany michael.strohmeier@uni-wuerzburg.de
[3]Prof. Sergio Montenegro is with the Chair of Aerospace Information Technology, Mathematics and Computer Science, University of Wuerzburg, Germany sergio.montenegro@uni-wuerzburg.de

978-1-7281-0778-3/19/$31.00 ©2019 IEEE

II. PROJECT DESCRIPTION

The system will be composed of two quad copters working in a master-slave principal to perform a formation flight to carry a net. The UAVs are complemented by a *Basestation* to command the two interceptor drones. By using the predicted trajectory of the target generated by the detection system of the MIDRAS project, we can find the optimal intersection point to catch the object with the net in mid air. To achieve this goal, the system needs to overcome several challenges, which will be discussed in more detail in the following chapters.

- Position and velocity of the target and also the interceptor drones need to be known at a good accuracy with small latencies
- Relative position and velocity between the UAV drone pair need to have a very high accuracy to perform a reliable formation flight to carry the net and catch the target
- The developed formation flight has to be stable and robust, especially at the moment of impact. This is very challenging since the UAVs are connected via a physical link
- Drones carrying the net need to be very agile and able to quickly adapt to changing weights and conditions
- The net which will be used to catch the target needs to be able to deal with different objects (i.e. quad copters, fixed wings)

III. STATE OF THE ART

Recently, a multitude of drone counter measures have been developed, both in military and commercial institutions as well as in educational and research facilities. A detailed summary of current commercial solutions is given in [4]. A classification of different approaches is introduced by Guvenc et al. [5]. The different counter measures can be divided into three categories, depending on their impact and level of control over the targeted drone: shoot down, complete take overs and catching methods.

In order to shoot down a hostile drone, it is not necessary to operate a UAV. Therefore, typical take down scenarios consider ground equipment which utilize mechanical or electromagnetic interference. In [6] a net bazooka is described to catch a drone from the ground. Multerer et al. [7] describe a signal jamming approach that interferes with piloting signals and causes a loss of control. Military solutions on the other hand rely on the use of laser [8] and machine guns [9]. Though the complete take down of a drone is efficient, there are restrictions to each of the methods presented above. While the net gun has a limited range, signal jamming is illegal and can interfere with other systems, too. The biggest drawback of take down methods is, that they cause an uncontrolled descent of the drone and therefore no complete elimination of its threats.

For drone take overs, two principle methods are distinguished. The first method is targeted towards manually or remote controlled UAVs and focuses on hacking the drone [10]–[12]. The second method is targeted towards autonomous or semi-autonomous flying drones that rely on GPS. So called spoofing attacks allow to feed wrong GPS information into the UAV system to force different flight maneuvers [13], [14]. Although drone take overs provide a complete threat elimination, they are not feasible for many scenarios. The variety of different drone protocols and their increasing security makes it difficult to hack a drone, especially if there is only a limited time frame. Following the trend towards autonomous drones using optical and inertial based navigation only, spoofing attacks might be ineffective. Furthermore, GPS spoofing attacks might effect other systems, such as autonomous cars, too.

Catching hostile drones in air gained popularity through videos by the Police of Tokio using a single drone with a net that caught another drone. However, single drones are restricted to a small net size and offer limited agility. In [15] a net shooting drone is presented that can either be used to take a drone down or to actually catch it. The difficulty within this approach is the precise and therefore slow alignment of the net gun. Klaussen et al. present a two drone configuration for recovering a fixed wing UAV on a known trajectory with a net [16], [17].

As seen there are many existing solutions to stop a possible harmful UAV, but each of the presented methods has its own drawbacks. To tackle some of these drawbacks and create a solution that is able to catch a UAV in a way that is safe and reliable, the *University of Wuerzburg* decided to create a new concept to catch drones which will be explained in the next chapter.

IV. CONCEPT

In Chapter II we already briefly discussed the main components of the system to be the interceptor drones working in a master-slave principle and a base station to communicate with the detection systems in the *MIDRAS* project and command the drones.

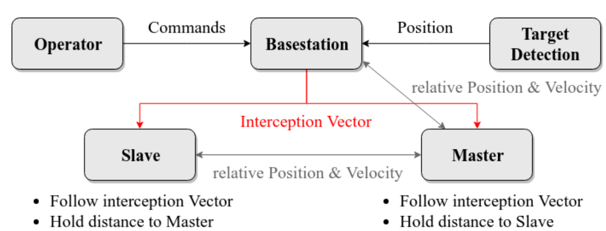

Fig. 2. Components of the Interception-System

In Figure 2 the individual components of the system are shown with their relation to each other. Since the detection is not part of our work we assume a reliable position and velocity estimate to be given by the target detection system. The components have the following tasks:

- **Basestation**: Receives commands from an optional *Operator* and the predicted trajectory of the target from the *Target Detection*. With these a interception vector is calculated, which is frequently updated to guide the interception drones to the optimal point of attack.
- **Master**: Coordinates the communication with the *Basestation* and the *Slave*. Controls its velocity to follow the given optimal interception vector. To do this it calculates the relative position and velocity to the *Basestation*.
- **Slave**: Controls its velocity to follow the given optimal interception vector and also hold its distance to the *Master* by using the relative positioning data calculated.

A. Platform

1) Hardware: The *MIDRAS* UAV platform is shown in Figure 3. The main frame is built from carbon fiber tubes which are connected via 3D printed clamps. Additional clamps allow an easy mounting of critical components such as the motors, batteries, Electronic Speed Controller (ESC) and flight controller. The 3D printed clamps are reinforced with carbon fiber inlays.

Fig. 3. The MIDRAS UAV platform.

The UAV propulsion is based on Components-Off-The-Shelf (COTS) as listed below:

- T-Motor MN4014 KV 400
- T-Motor 15x5.0 Carbon Rotors
- T-Motor S24A 600Hz ESC

The custom made flight controller used on the platform was developed at the *University of Wuerzburg* and is shown in Figure 4.

The flight controller consists of two stacked boards. The bottom board is a UDOO Neo Extended based on the i.MX6 SoloX heterogeneous dual-core processor. The top board is custom made and provides interfaces for commonly used drone sensors, such as optical flow sensors, range lidar systems, long range communication modules or thermal cameras. Additionally, the following sensor suit is available:

- Redundant 9-DoF Inertial Measurement Unit (IMU)
- Barometric Pressure Sensor
- Ultra-Wide Band (UWB) Modul
- Global Navigation Satellite System (GNSS) Receiver

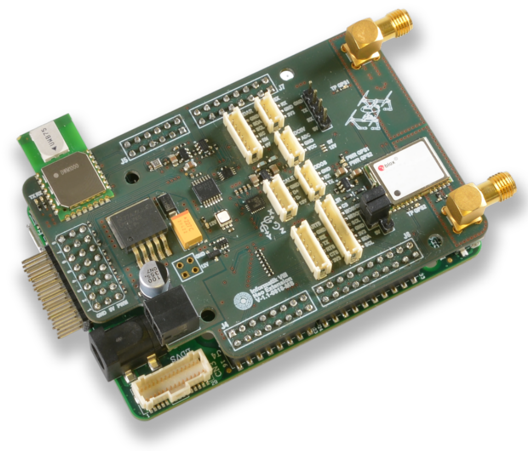

Fig. 4. The MIDRAS UAV flight controller.

2) Software: Utilizing the heterogeneous dual-core architecture, the workload for all in-flight tasks is distributed according to their real-time constraints (see Figure 5). The Real-time Onboard Dependable Operating System (RODOS) [18] is running on the Cortex-M4, providing the necessary capabilities for hard real-time tasks with latencies of milliseconds and lower, such as UAV state estimation and control as well as timing critical tasks like UWB communication and range measurements. Tasks that are computationally expensive are executed on the Cortex-A9. The Cortex-A9 provides a Linux environment and runs the Robot Operating System (ROS) [19]. The two cores communicate with each other using shared memory and the Remote Processor Messaging (RPMsg) protocol [20].

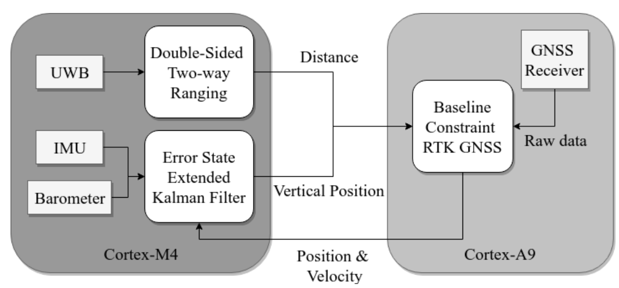

Fig. 5. Workload distribution on the dual-core architecture

B. Relative Positioning

The relative position between the two UAVs is estimated using two loosely-coupled Extended Kalman Filter (EKF). The first filter is an error state EKF using quaternions based on the work published by Sola [21]. It is implemented on the Cortex-M4 and updates its state estimates at up to 1 kHz. The second filter runs on the Cortex-A9 and combines GNSS raw code and carrier observation at 10 Hz from GPS, Glonass and Galileo satellites with UWB range measurements between the two UAVs and with the state estimate output of the first EKF. The state of the first EKF is given by:

$$\boldsymbol{X}_{M4} = (\boldsymbol{p}, \boldsymbol{v}, \boldsymbol{q}, \boldsymbol{b}_a, \boldsymbol{b}_\omega, \boldsymbol{b}_m, K)^T , \tag{1}$$

where $\boldsymbol{p}$ and $\boldsymbol{v}$ are the UAV's position and velocity in the navigation frame, $\boldsymbol{q}$ is the current attitude quaternion, $\boldsymbol{b}_i$ with $i \in [a, \omega, m]$ are the bias values for the accelerometer, gyroscope and magnetometer, respectively, and K is the IMU temperature.

In the Cortex-M4 EKF, the accelerometer and the gyroscope are used for filter propagation, regarding the position and the attitude, respectively. The position estimates are corrected using the output of the second EKF running on the Cortex-A9. The roll and pitch estimates are corrected based on a low-pass estimate of the gravitational vector using the accelerometer, while the yaw component is corrected using magnetic north estimates. Additional information from a barometric pressure sensor is used for further altitude corrections.

The Cortex-A9 EKF is based on the Open Source project RTKLIB [22]. The state of this second EKF is given by:

$$\boldsymbol{X}_{A9} = \left(\boldsymbol{r}_s, \boldsymbol{v}_s, \boldsymbol{B}^j_{sm}\right)^T, \tag{2}$$

where $\boldsymbol{r}_s$ and $\boldsymbol{v}_s$ are the slave drone position and velocity, respectively, and $\boldsymbol{B}^j_{sm}$ are single difference carrier-phase biases for common satellite system observations $j \in \{0, 1, ..., n_{obs}\}$ of the slave and the master drone. Given a master position $\boldsymbol{r}_m$ and velocity $\boldsymbol{v}_m$, the relative position $\boldsymbol{r}_{sm}$ and velocity $\boldsymbol{v}_{sm}$ between the two UAVs are simply given by the differences:

$$\boldsymbol{r}_{sm} = \boldsymbol{r}_s - \boldsymbol{r}_m, \;\; \boldsymbol{v}_{sm} = \boldsymbol{v}_s - \boldsymbol{v}_m \tag{3}$$

The master's position and velocity can be estimated either by single positioning using the master's raw observations or by estimating a RTK based GNSS solution with respect to a known base station.

Additionally to the native RTKLIB implementation, hard baseline constraints are introduced during integer ambiguity resolution based on Double-Sided Two-Way Ranging (DS-TWR) UWB range measurements [23] d_{UWB} and the vertical height difference between the two drones Δh:

$$|\boldsymbol{r}_{sm}| = d_{UWB} \tag{4}$$

$$|r_{sm_z}| = \Delta h \tag{5}$$

C. Interception Vector

Concerning the safety of the object to protect, it is very important that the unauthorized UAV is captured as fast and securely as possible. To reach these goals, we need to calculate a trajectory for the interception drones to follow, which we will call interception vector $\boldsymbol{v}_i$. This interception vector needs to be updated frequently and lead the drones to its interception point. Since we do not know the objectives and capabilities of the invading UAV we are assuming a waypoint flight with straight lines between the points, which is most common in modern commercially available UAVs.

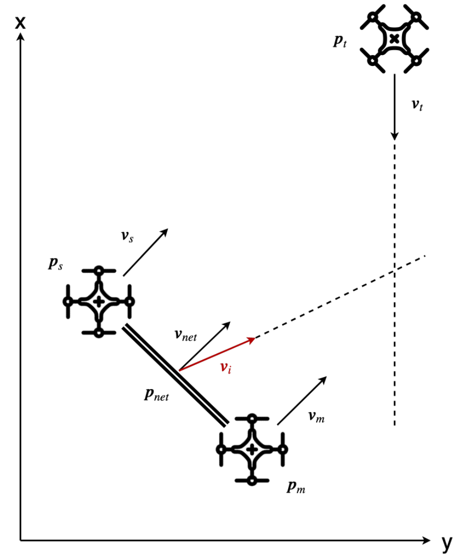

Fig. 6. Basic concept of calculating the interception vector

The basic concept of solving this optimization problem is shown in a 2-dimensional format in Figure 6. We use the trajectory of the target given by its 3-dimensional position and velocity vector $\boldsymbol{p}_t$ and $\boldsymbol{v}_t$, as well as the trajectories of the interception drones $\boldsymbol{p}_m$, $\boldsymbol{v}_m$ and $\boldsymbol{p}_s$, $\boldsymbol{v}_s$ of the master and slave drone, respectively to find the optimal interception point. By simplifying the task to the relevant information we get an optimization problem. First we need to find the trajectory $\boldsymbol{p}_{net}$, $\boldsymbol{v}_{net}$ and heading θ_{net} of the center of the quadratic net with the side length l_{net} carried by the drones.

$$\theta_{net} = atan2\left(p_{m_x} - p_{s_x}\;,\; p_{m_y} - p_{s_y}\right) \tag{6}$$

$$\boldsymbol{p}_{net} = \boldsymbol{p}_m + \frac{l_{net}}{2}\begin{bmatrix} sin\left(\theta_{net}\right) \\ cos\left(\theta_{net}\right) \\ -1 \end{bmatrix} \tag{7}$$

$$\boldsymbol{v}_{net} = \frac{\boldsymbol{v}_m + \boldsymbol{v}_s}{2} \tag{8}$$

With this we can calculate the velocity vector $\boldsymbol{v}_i$ by optimizing the intersection point $\boldsymbol{b}_i$ of the predicted trajectory of the target and the optimal trajectory of the net for a minimal time $\boldsymbol{t}_i$ within a given time interval T and a maximum speed v_{max}:

$$\min_{t_i \in T} t_i \tag{9}$$

subject to

$$\left|\boldsymbol{v}_i(t_i)\right| = \left|\frac{\boldsymbol{p}_t + \boldsymbol{v}_t \cdot t_i - \boldsymbol{p}_{net}}{t_i}\right| \leq v_{max} \tag{10}$$

If we can not find a solution to this problem, this means, that we are not able to intercept the target in the given time interval T for different reasons, i.e. the target is moving in the opposite direction as our protected perimeter lies. The interception drones will stay in the air and wait for a possible interception point or until they are commanded to come back to base. On the other hand, if there is a solution to the problem we also need to make sure that the rotation of the net is orthogonal to the targets trajectory at the time of impact. To achieve this we are updating our desired net heading to be orthogonal to an imaginary line between the target and the current net position.

$$\theta_i = atan2\left(p_{t_x} - p_{net_x}\ ,\ p_{t_y} - p_{net_y}\right) - \pi/2$$

D. UAV Control

To catch a UAV in mid air, the two interceptor drones will work together to span a net between them. For this task they will use the high precision position and velocity data gained through the relative positioning algorithm described in Chapter IV-B. The controller of the UAVs working in a master-slave principle is designed in a passive way as described in [24]: Each drones inner loop velocity controller is using its own measurements to meet the desired velocity, which is controlled by an outer loop position controller using all available information about the other drone and the target. Therefore both drones will have two main objectives to control.

- **Follow the interception vector:** The interception vector calculated by the base station will be transmitted to both master and slave. Each drone needs to control its velocity to follow this vector as fast and accurate as possible. The vector will be updated frequently to guide the interceptor drones to the optimal point of impact.
- **Keep in formation:** To keep the net spanned and avoid possible collisions, especially on impact, both drones need to control their relative distance in 3-dimensional space to a given distance $\boldsymbol{d}_{rel}$. The relative distance is dependent on the length of the net l_{net} and the desired orientation θ_i.

$$\boldsymbol{d}_{rel} = l_{net} \begin{bmatrix} sin\left(\theta_i\right) \\ cos\left(\theta_i\right) \\ 0 \end{bmatrix} \tag{11}$$

Both of these control objectives will result in a combined desired velocity $\boldsymbol{v}_{des}$ the drones need to follow. To control the velocity the system will be using a cascaded PID-controller. The whole control structure for the *Slave* is visualized in Figure 7.

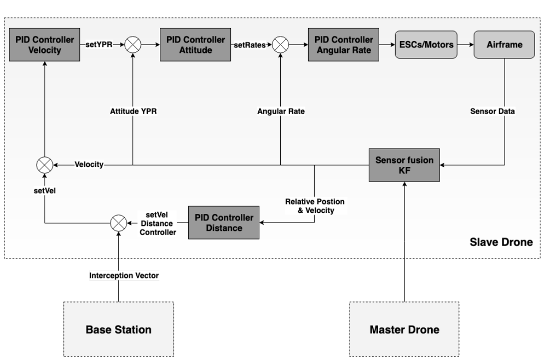

Fig. 7. Control Structure for the *Slave*

When catching the target in the net, both drones will not only experience great forces which will pull them physically together, but they will also immediately need to adapt to the changing weight and agility of the whole system. Since simple PID-Controllers can not adapt to such changes fast enough, the system will be working with a Model Reference Adaptive Controller (MRAC) to be able to adjust its agility on the fly as fast as possible as shown in [25].

V. First Results

A. Impact On Net

To get a better understanding of the forces in play when catching UAV in mid air with a net, a drone was flown into a net in an controlled environment. The used drone had a mass of $m = 1050$ g and a diameter of $d = 450$ mm To capture the trajectory of the drone an optical tracking system was used which captured the flight at a rate of 50 Hz. This data was used to plot the whole trajectory of the flight into the net in Figure 8. After the impact the net starts oscillating around the point of impact.

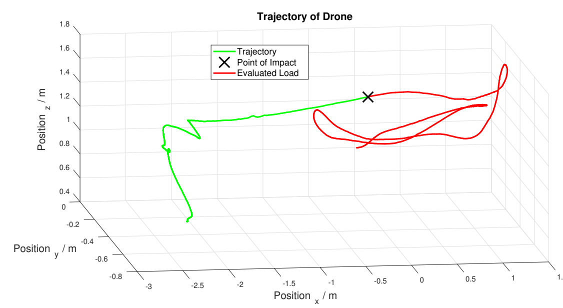

Fig. 8. Trajectory of the drone flown into the net

The OptiTrack data could also be used to find the maximum velocity at the time of impact $v_{impact} = 6.4\,\mathrm{ms}^{-1}$. Additionally load cells on the suspension point of the net were used to get data of the forces acting on the drones at a rate of 100 Hz, which can be seen in Figure 9. A lot of the impact energy is transfered into oscillations in the net and the maximum loads measured were about 2.5 times the mass of the drone.

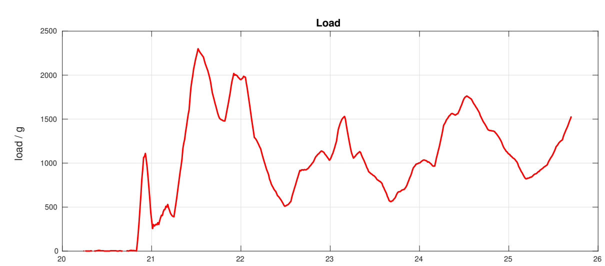

Fig. 9. Load at the point of impact

B. Formation Flight

To implement the first algorithms for the formation flight of the master-slave drone pair, the positions were gained by using an optical tracking system indoor. This has the advantage of being a proven system with high accuracy in the position data. Also, the *Flight Cage* of the department for aerospace information system could be used to have a controlled test environment. The only disadvantage is the decreased space of only 6x6x6 cubic meter. Therefore a smaller quadrocopter-frame of only 20 cm in diameter was developed and used to test the first formation flights indoor. Figures 10 and 11 show flights of the master-slave system physically connected carrying a net of 1x1 meter, where the slave was commanded to hold its relative position to the master at $\boldsymbol{p}_{rel} = \begin{bmatrix} 0 & 1 & 0 \end{bmatrix}^T$ m.
In Figure 10 the drone pair was commanded to reach and hold a position to proof the stability of the system. The plots show the position of both drones as well the calculated distance in all axis in meter. After take off the slave has a small delay to the master, which can especially be seen in the Z-distance. After this delay is overcome, there are light oscillations by the slave as it orients itself to the desired position $\boldsymbol{p}_{rel}$ next to the master.

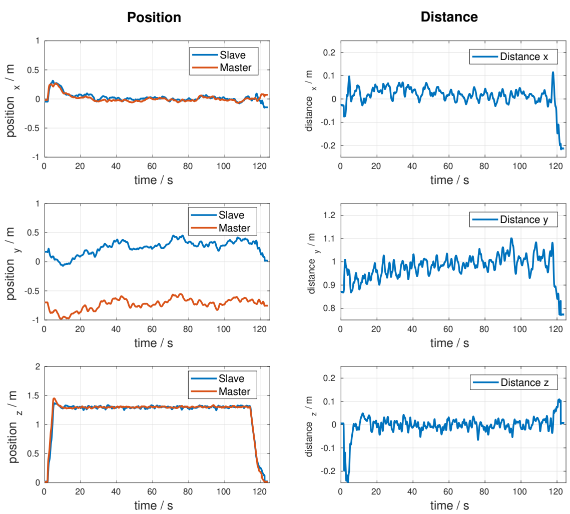

Fig. 10. Formation Flight in 3 axis

In Figure 11 small maneuvers in all three dimensions were flown. It can be seen, that the delay between the drones is almost zero, so that the maneuvers are flown simultaneously by the drones. At $t = 56$ s the slave was physically disturbed by pushing it in the negative x direction to test the robustness of the solution. After a short deviation in the position the slave comes back into formation after 10 seconds.

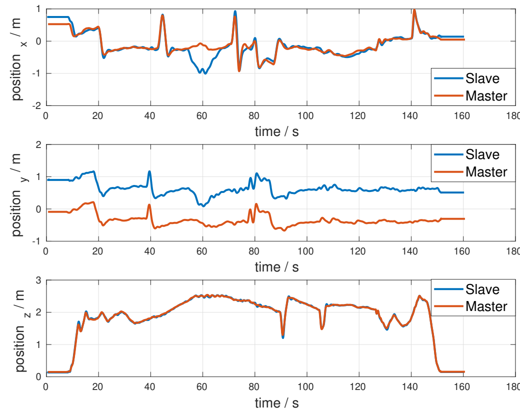

Fig. 11. Formation Flight in 3 axis

VI. CONCLUSIONS

In this paper we showed our concept for developing a UAV system capable of catching possibly harmful drones. As seen in Chapter III, there is currently no existing method of stopping UAVs in both a safe and reliable way, like our concept is proposing. First indoor flights have shown promising results for the idea of using a master-slave principle to carry a net with two drones. In the next step the proposed solution for an highly accurate outdoor relative positioning needs to be further evaluated and integrated into the system for a reliable outdoor formation flight.

ACKNOWLEDGMENT

This research has been funded by the Federal Ministry of Education and Research of Germany in the framework of MIDRAS (project number 13N14315).

REFERENCES

[1] Bundesverband der Deutschen Luftverkehrswirtschaft e.V. Analyse des deutschen Drohnenmarktes. Online: `https://www.bdl.aero/de/publikation/analyse-des-deutschen-drohnenmarktes`. Accessed: 2019-04-26.

[2] Bundesministerium fuer Bildung und Forschung. Sicherheitsforschung. Online: `https://www.sifo.de/en/index.html`. Accessed: 2019-04-26.

[3] Bundesministerium fuer Bildung und Forschung. Projektumriss MIDRAS: Mikro Drohnen Abwehr System. Online: `https://www.sifo.de/files/Projektumriss_MIDRAS_C3.pdf`. Accessed: 2019-04-26.

[4] Arthur Holland Michel. *Counter-Drone Systems*. Center for the Study of the Drone at Bard College, 2018.

[5] Ismail Guvenc, Farshad Koohifar, Simran Singh, Mihail L Sichitiu, and David Matolak. Detection, Tracking, and Interdiction for Amateur Drones. *IEEE Communications Magazine*, 56(4):75–81, 2018.

[6] Corrin Gray. Net Launching Tool Apparatus, September 26 2002. US Patent App. 09/814,527.

[7] Thomas Multerer, Alexander Ganis, Ulrich Prechtel, Enric Miralles, Askold Meusling, Jan Mietzner, Martin Vossiek, Mirko Loghi, and Volker Ziegler. Low-cost jamming system against small drones using a 3d mimo radar based tracking. In *2017 European Radar Conference (EURAD)*, pages 299–302. IEEE, 2017.

[8] Raytheon. Beam On - A Wide Range of Counter-Drone Technologies Comes of Age. Online: `https://www.raytheon.com/news/feature/beam-on`. Accessed: 2019-04-26.

[9] Leonardo DRS. Reconfigurable integrated-weapons platform. Online: https://www.leonardodrs.com/media/6357/reconfigurable-integrated-weapons-platform_riwp_500-1176_0917.pdf. Accessed: 2019-04-26.

[10] Johann-Sebastian Pleban, Ricardo Band, and Reiner Creutzburg. Hacking and securing the ar. drone 2.0 quadcopter: investigations for improving the security of a toy. In *Mobile Devices and Multimedia: Enabling Technologies, Algorithms, and Applications 2014*, volume 9030, page 90300L. International Society for Optics and Photonics, 2014.

[11] Nils Rodday. Hacking a professional drone. *Slides at www. blackhat. com/docs/asia-16/materials/asia-16-Rodday-Hacking-A-Professional-Drone. pdf*, 2016.

[12] Kyle Wesson and Todd Humphreys. Hacking drones. *Scientific American*, 309(5):54–59, 2013.

[13] Daniel P Shepard, Jahshan A Bhatti, Todd E Humphreys, and Aaron A Fansler. Evaluation of smart grid and civilian uav vulnerability to gps spoofing attacks. In *Radionavigation Laboratory Conference Proceedings*, 2012.

[14] Andrew J Kerns, Daniel P Shepard, Jahshan A Bhatti, and Todd E Humphreys. Unmanned aircraft capture and control via gps spoofing. *Journal of Field Robotics*, 31(4):617–636, 2014.

[15] Mohammad Rastgaar Aagaah, Evandro M Ficanha, and Nina Mahmoudian. Drone Having Drone-Catching Feature, 2017. US Patent App. 15/332,170.

[16] Kristian Klausen, Thor I Fossen, and Tor Arne Johansen. Autonomous Recovery of a Fixed-Wing UAV Using a Net Suspended by Two Multirotor UAVs. *Journal of Field Robotics*, 35(5):717–731, 2018.

[17] Kristian Klausen, Jostein Borgen Moe, Jonathan Cornel van den Hoorn, Alojz Gomola, Thor I Fossen, and Tor Arne Johansen. Coordinated Control Concept for Recovery of a Fixed-Wing UAV on a Ship Using a Net Carried by Multirotor UAVs. In *2016 International Conference on Unmanned Aircraft Systems (ICUAS)*, pages 964–973. IEEE, 2016.

[18] RODOS - Real-time Onboard Dependable Operating System. Online: https://en.wikipedia.org/wiki/Rodos_(operating_system).

[19] About ROS. Online: http://www.ros.org/about-ros/. Accessed: 2019-04-26.

[20] Petr Lukas. RPMsg Messaging Protocol. Online: https://github.com/OpenAMP/open-amp/wiki/RPMsg-Messaging-Protocol. Accessed: 2019-04-26.

[21] Joan Sola. Quaternion kinematics for the error-state kf. *Laboratoire dAnalyse et dArchitecture des Systemes-Centre national de la recherche scientifique (LAAS-CNRS), Toulouse, France, Tech. Rep*, 2012.

[22] Tomoji Takasu. Rtklib: Open source program package for rtk-gps. *Proceedings of the FOSS4G*, 2009.

[23] DecaWave. APS013: The implementation of two-way ranging with the DW1000 . Online: https://www.decawave.com/sites/default/files/aps013_dw1000_and_two_way_ranging_v2.2.pdf. Accessed: 2019-04-26.

[24] He Bai, Murat Arcak, and John Wen. *Cooperative control design: a systematic, passivity-based approach*. Springer Science & Business Media, 2011.

[25] Afaque Khan and MN Shanmukha Swamy. Modified mrac based on lyapunov theory for improved controller efficiency. In *2016 International Conference on Automatic Control and Dynamic Optimization Techniques (ICACDOT)*, pages 989–995. IEEE, 2016.