



DR 2.1: Baseline teleoperation manipulation and waypoint navigation capabilities

Petter Ögren*, Mike Bosse†, Rainer Worst‡, Michael Achtelik§ and the TRADR Consortium

*KTH - Royal Institute of Technology, Stockholm, Sweden

†ETH Zürich - Autonomous Systems Lab, Zurich, Switzerland

‡Fraunhofer IAIS, Sankt Augustin, Germany

§Ascending Technologies GmbH, Krailing, Germany

<petter@kth.se>

<i>Project, project Id:</i>	EU FP7 TRADR / ICT-60963
<i>Project start date:</i>	Nov 1 2013 (50 months)
<i>Due date of deliverable:</i>	Month 14
<i>Actual submission date:</i>	15th of February 2015
<i>Lead partner:</i>	KTH
<i>Revision:</i>	final
<i>Dissemination level:</i>	PU

This document (D2.1) describes the work of providing the TRADR robots (UAVs and UGVs) with baseline teleoperation manipulation and waypoint navigation capabilities. These capabilities will enable the robots to perform basic search and rescue missions in both synchronous (teleoperation) and asynchronous (waypoint navigation) modes. In the user tests, the more advanced functions will then be evaluated against these baseline capabilities.

1	Tasks, objectives, results	5
1.1	Planned work	5
1.2	Actual work performed	5
1.2.1	Task 2.1 Standard Teleoperation of UGVs	5
1.2.2	Task 2.4 UGV Waypoint Following	6
1.2.3	Task 2.4 UAV Waypoint Following	9
1.3	Relation to the state-of-the-art	14
	References	16
2	Annexes	19
2.1	Beul et al. (2014), “Nonlinear Model-based Position Control for Quadrotor UAVs”	19
2.2	Wilkes (2015), “3D Navigation for UAVs in GPS denied environments (master’s thesis)”	19
2.3	Wilkes (2015), “3D Navigation for UAVs in GPS denied environments (draft paper)”	20
2.4	Konze (2014), “Laser-based obstacle avoidance for autonomous multicopters”	20
2.5	Arya (2015), “Autonomous UAV Navigation in Unknown Indoor Environment using Onboard Light Weight Camera.”	20
A	Nonlinear Model-based Position Control for Quadrotor UAVs	22
B	3D Navigation for UAVs in GPS denied environments (master’s thesis)	28

Executive Summary

This report describes work towards providing the TRADR robots, Unmanned Aerial Vehicles (UAVs) and Unmanned Ground Vehicles (UGVs) with basic motion capabilities, in terms of teleoperation and waypoint navigation. These capabilities will then serve as baselines for evaluation of more advanced capabilities.

Formally, the report constitutes deliverable (D2.1), the first year deliverable of WP2 (persistent models for acting). The aim of WP2 is to endow the TRADR robots with motion capabilities. Applying a user centric design, these capabilities will then be extended each year of TRADR, and evaluated by end users comparing the new features to the previous ones. During the first year, WP2 included tasks T2.1 (Standard teleoperation mode) and T2.4 (Waypoint following).

In task T2.1 (Standard teleoperation mode) the UGV was provided with teleoperation functionality for the mobile base as well as the newly integrated robot arm. This functionality is indeed very standard, allowing the UGV operator to issue velocity commands using a gamepad, to the two tracks as well as the joints of the arm. The arm teleoperation will be tested by the end users during the next TRADR Joint-Exercise.

Task T2.4 (Waypoint following) involved both the UGVs and the UAVs. For the UGVs, a D*-Lite planner from NIFTi was upgraded to the latest operating system distributions and tested during the last TRADR Joint-Exercise, in Pisa 2014. One request of the end users was faster planning times. For the UAVs, waypoint following in obstacle free environments is in place, but waypoint following in the presence of obstacles is a much harder problem. Problems regarding trajectory optimization, obstacle avoidance, position maintenance in case of GPS signal loss, and visual corridor following were investigated, but no results were tested by end users.

Role of teleoperation and waypoint navigation in TRADR

Teleoperation and waypoint navigation represent the baseline motion capabilities of the UGVs and UAVs in the TRADR setup. During the project, more advanced functionalities will be developed. In order to assess the performance and utility of new functionalities, we have to be able to compare them to some baseline functionalities. These are the pure teleoperation and waypoint navigation described here.

Contribution to the TRADR scenarios and prototypes

Controlling the robots is a fundamental part of all TRADR scenarios. Baseline teleoperation and waypoint navigation gives the end users a choice between synchronous (teleoperation) and asynchronous (waypoint navigation) control of the UGVs and UAVs. These choices span the space of control possibilities and will later be complemented with other options, that can then be compared to the two baseline ones.

During the TRADR Joint-Exercise in Pisa 2014, the UGV base was teleoperated, but not the arm, since it was not integrated yet. The waypoint following of the UGV was also used. The UAVs were teleoperated, and used in waypoint following mode to automatically collect data for the creation of a 3D model of the disaster area. UAV waypoint following in case of GPS signal loss, or in the presence of obstacles, is however still an open research area.

1 Tasks, objectives, results

In this section we will give a somewhat detailed description of the work performed, but first also give a brief description of the work planned, to show that we are in line with the overall project.

1.1 Planned work

The work described in this report (D2.1) aims towards Milestone MS2.1, and was performed within the scope of Tasks T2.1 (Standard teleoperation mode) and T2.4 (Waypoint following). D2.1 and MS2.1 are planned and described in the TRADR Division of Work (DoW) as follows.

D2.1: Baseline teleoperation manipulation and waypoint navigation capabilities: Describes the analysis and algorithms for achieving MS2.1. At the end of Y1, WP2 will provide basic motion and manipulation capabilities to the TRADR platforms. These capabilities will enable them to perform the missions and will be evaluated in the user tests of T5.1. This milestone is achieved through Task 2.1 and 2.4, and documented in this deliverable.

Milestone MS2.1: Baseline teleoperation manipulation and waypoint navigation capabilities (M14) At the end of Y1, the main result of WP2 will be to provide the initial motion and manipulation capabilities to the TRADR platforms. More specifically, we will first demonstrate manipulation capabilities of platforms based on teleoperation in line with the work described in in Task 2.1. This is necessary to achieve a default mode in cases where autonomy is neither desired nor possible. The second part of the milestone is related to the platforms' ability to plan and execute paths in full 3D environment as outlined in Task 2.4. By reaching the milestone it will be demonstrated that each platform is able to autonomously follow a set of predefined waypoints. To evaluate the results of the milestone, we will use the empirical benchmarks to be created in T5.1. These benchmarks focus primarily on navigation and manipulation, and include controls as for the level of overt/covert-ness of responsibility transfer, and help requests. The results of this milestone will be documented in deliverable DR2.1.

1.2 Actual work performed

The work performed in the areas of UGV teleoperation, as well as waypoint navigation for UGVs and UAVs is presented below.

1.2.1 Task 2.1 Standard Teleoperation of UGVs

In this section we will describe how basic teleoperation of the UGV platform (both manipulator arm and tracks) has been implemented.

We equipped the UGV mobile platform with a Kinova JACO Research Edition arm¹. It is a lightweight robot arm with 6 Degrees of Freedom, equipped with 3 underactuated fingers. The arm can extend up to 90 cm and lift 2 Kg at mid-range, and 1 Kg when fully extended. A wired remote control is included, and there is support for controlling the arm using a computer. Kinova Robotics provides APIs which offer basic functionalities, including Robot Operating System (ROS) support. Teleoperation is performed by passing ROS messages. For more details on the choice of robot arm, see D6.1.

Baseline teleoperation of the mobile platform is already possible with the legacy NIFTi-system, using a gamepad or keyboard. The implemented control mode is the classical *Tank Control* where the user sets velocity and rotation of the mobile platform.

We implemented baseline teleoperation of the arm using the Kinova ROS API, with a gamepad controlling the arm in either a cartesian coordinate system or using joint space control. The user can take advantage of either the ladybug camera or the estimated 3d model of the UGV to have a visual feedback on the relative position of the arm while controlling it. The user also has the possibility to control the fingers and performing basic manipulation, e.g. grasping an object. The current functionality for arm teleoperation provides three more basic features: automatic return to home position by pressing a button, safe emergency stop of the arm, and estimation of the forces acting on the tool, using motor current measurements.

1.2.2 Task 2.4 UGV Waypoint Following

Waypoint following is a low-level functionality of both Unmanned Air Vehicles (UAVs) and Unmanned Ground Vehicles (UGVs). It addresses the necessary aspects of autonomously planning and executing a path from a given starting configuration to a goal configuration. The application for the first year is a go-home routine, which enables the robot to autonomously return to a safe home position after the completion of a mission.

Overview In NIFTi a D*-Lite based planning algorithm was developed and tested for the UGV [5]. A schema-block overview of this algorithm is illustrated in Figure 1. It plans the robot pose on a static map derived from the recorded point cloud map. Before computing a path, the sparse point cloud obtained from the localization and mapping module, based on Iterative Closest Point algorithm, needs to be densified to achieve a structural representation which describes the presence and orientation of surfaces, at a grid resolution of 5cm. This densification is performed by a tensor voting algorithm described in [16]. Once this desired 3D representation has been

¹Kinova Robotics: <http://kinovarobotics.com/products/jaco-robotics/>

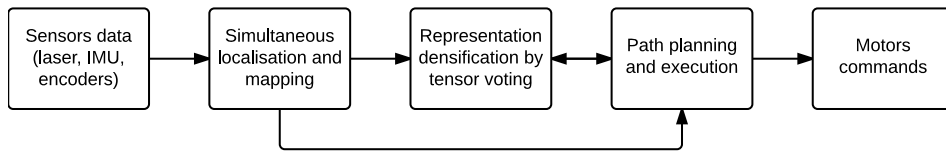


Figure 1: UGV waypoint following system overview.

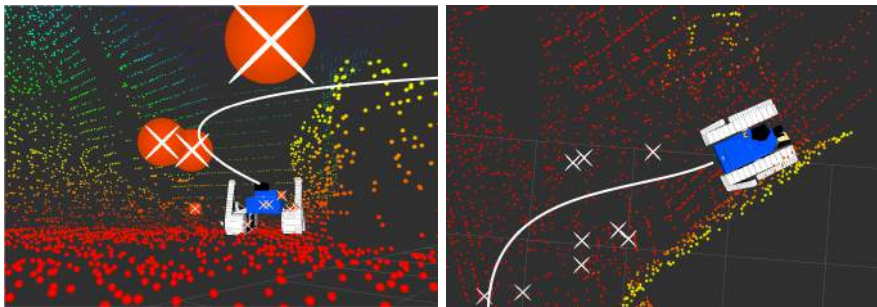


Figure 2: Shadow points (white crosses) created by flippers.

achieved, possible poses and transitions in between are determined. Based on this connectivity information, D*-Lite is responsible of finding the most adequate sequence of poses from a starting position to the goal. This path gets decorated with a given set of flipper configurations and robot orientation therefore planning for all of the robots degrees of freedom. During path execution, the path is frequently re-planned in order to react if the robot position diverges from the initially planned trajectory. The D*-Lite algorithm enables efficient re-planning which is a desirable attribute in these situations. In the course of TRADR the planner was successfully upgraded to the most recent robot operating system distribution (ROS Indigo Igloo) and refined in its performance.

TRADR Joint-Exercise experiences During the TRADR Joint-Exercise in Pisa, the main focus was to assess the performances of this initial system, identifying advantages, drawbacks and possible area of improvement. It was also deployed on most of the consortium's UGVs.

The planner performance is closely coupled to the quality of the point cloud maps and its densified representation. Outliers and artifact points are likely to decrease the planner's success and increase computational efforts. One source of outlier points is from the shadowing of the UGV's flippers, as illustrated in Figure 2. The shadow points filtering module parameters were adjusted in order to obtain better filtering results, therefore enhancing the planning capabilities. Another issue is the presence of ground patches

where laser points failed to be registered, again limiting the space of valid paths. To overcome these limitations in a quick fashion, the settings of the planner were relaxed to accept greater margins of outliers and deviations from the expected path. This led to qualitatively higher performance of the system for the cost of less accurate way-point following.

A deficiency which was clearly identified by the end-users was the robot switching to an idle state lasting for several seconds, occurring when the planner failed at re-planning its path. This delay was within the D*-Lite search and occurred when the planner was searching for a solution and was ultimately going to conclude that there are no solution. It then took several seconds before it could realize this and move forward to the next step. In this idle state, it was hard to understand why the robot failed to re-plan.

Planning and Flipper Control During field tests the planner was combined with the Adaptive Traversability Flipper control described in [22]. However, the performance of the whole system decreased in challenging areas since both modules yielded different flipper configurations, resulting in frequent re-planning as the planner did not reach its expected configurations. As a result, the planner had to stop execution and re-plan a new trajectory much more often than usual. In order to move forward in this research direction, the planner will need to be explicitly informed about a separate flipper control module.

Evaluations End user evaluations revealed a number of drawbacks towards performance and user friendliness. It was reported that the planning time (in the range of 1 second per meter of distance between start and goal position) for initial paths was too slow. The robot was at first moving extremely slow which was addressed by a different set of control parameters for the planner. The graphical user interface of the planner was also reported to be only usable by experienced users. Furthermore, a user-friendly information about the internal planner state is desirable. Ultimately the reliability of the way-point following needs improvement in the next phase of TRADR.

Future considerations Regarding UGV waypoint following, research will be conducted in these areas; Upgraded hardware on the UGV as well as reuse of planner state and augmentation of maps with planning hooks shall enable the robot to perform on-board map segmentation and path planning. This has not been possible in NIFTi due to limited on-board computation capabilities of the robots. Furthermore an investigation of close integration between mapping and path planning will be conducted. Another lead will be a serialization of the path planning problem into small, potentially reusable chunks. The limitation of path planning on static maps will be addressed

in the next years. The robots will be enabled to plan on dynamic maps and potentially travel into previously unexplored regions.

1.2.3 Task 2.4 UAV Waypoint Following

As for the UGV also the UAV should be able to follow given waypoints automatically, as an alternative to teleoperation.

Overview In NIFTi the UAV was manually teleoperated to acquire laser and image data from the scenes. The commands from the UAV operator regarding how to explore the scenario were transmitted via walkie-talkie to the UAV pilot, who steered the UAV in response to the UAV operator requests. Compared to the UGV, the UAV has to deal with some additional challenges:

- significantly limited energy resources, max flight time of about ten minutes
- in standard teleoperation, the pilot always needs to maintain a clear line of sight to the UAV, thus, standard teleoperation through the OCU is not possible
- quick reaction time is necessary to safely steer the UAV
- quick obstacle detection and avoidance needed to enable automatic waypoint navigation
- when using a horizontally mounted 2D laser scanner, the creation of 3D models is only possible after extensive altitude changes so-called *elevator flight*
- only limited payload available

Furthermore, compared to the UGV, the UAV is very sensitive to remote control signal loss. To handle remote control signal loss several strategies for the UAV are conceivable:

- land slowly
- stay where you are (hover)
- go home (or to last point of remote connection)
- keep on exploring unknown environment autonomously

For a save strategy it is crucial that the parameters to achieve the goal are balanced carefully to avoid a crash.

Below, we will discuss a number of issues related to UAV waypoint following in some more detail.

The position controller The NIFTi-UAV, which was custom-built for the NIFTi project based on the Mikrokopter toolkit, serves as a mobile sensor platform and works in cooperation with the humans involved.

Its main task when used in an earthquake scenario is to gain situation awareness during the execution of a rescue operation and to share it with the human rescuers. For this task, large-scale autonomy is necessary and the UAV must be capable to reach and stay at specified waypoints.

Because the default PID-controller provided by Mikrokopter did not work adequately for the NIFTi-UAV, we designed and implemented our own closed loop position controller, which is described in [4] (Annex Overview 2.1). The approach proposed in the underlying Master's thesis initially models the kinematics of the UAV and develops a custom control algorithm based on the model, which is experimentally modified to match the behaviour of the real robot.

It appeared that a nonlinear cascaded controller delivers optimal results navigating from a given start location A to a given goal location B.

Optimizing UAV trajectories through a set of waypoint Performing GPS waypoint flying, a huge difference between state of the art systems and the novel control systems developed by ASC in previous projects and optimized in TRADR is the way of handling the waypoints. Flying point to point or controlling the trajectory in between. This concept has been taken even one step further: Trajectories are optimized throughout all waypoints and an optimal path taking into account the limitations of the UAV is calculated. For the first T-Jex this feature was combined with a simple editor in the control station enabling a manual pilot to simply define a matrix shape search pattern over a building. Doing so, precise photogrammetry information can be collected in a very short time, and used to compute a complete 3D model of the environment. For more details, please refer to D6.1.

Obstacle avoidance Another thesis [10] (Annex Overview 2.4) covers the development and implementation of a method to avoid obstacles in the path of the NIFTi-UAV during automatic flight, while following GPS waypoints. In the proposed method, the surroundings of the UAV are scanned via a fixed mounted laser scanner. If any obstacles in the current flight direction are found, the UAV stops in front of the obstacle.

Due to the fixed mounting of the laser scanner, the scanned section of the environment is very narrow. In order to enlarge this section, the UAV performs an elevator maneuver during which the UAV descends to a certain point and then returns to its original altitude. While performing this maneuver, a map of the UAVs surroundings is created. Using this map, an evade-point is calculated and set as the next waypoint. When this evade-point is reached, the normal automatic waypoint flight is resumed until the

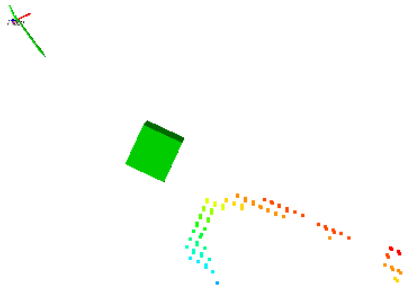


Figure 3: Safe position in front of an obstacle

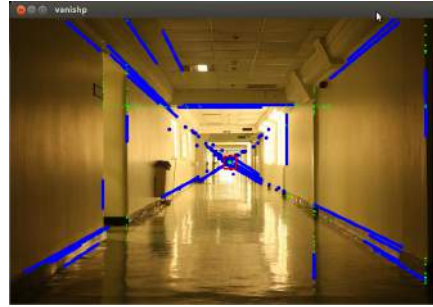


Figure 4: Corridor vanishing point

next obstacle appears.

The sample image in Figure 3 shows a detected obstacle as a point cloud. The green box is a marker for the waypoint in front of the obstacle, where the UAV has to stop in order to avoid a collision. The green arrow indicates the current flight direction.

Low level controller for position maintenance and waypoint following For all waypoint, obstacle avoidance and path planning tasks, there is one more important difference between UAVs and UGVs: While a UGV can simply hold position, a UAV will immediately start drifting away if it loses a position reference signal e.g. GPS. Therefore UAV waypoint following or simple hovering on one spot is very time critical. All algorithms need to run real time at a high update rate. Especially computationally intense tasks, like vision based navigation, can lead to significant delays and unstable behavior. Furthermore delays caused by network transmissions cannot be accepted on the UAV. Switching to Ascending Technologies UAVs in TRADR, this problem was solved by integrating one more control layer. The UAVs' Flight Control Unit (FCU), which is based on microprocessors operating real-time with an update rate of 500-1000 Hz, controls position hold and waypoint following directly in GPS conditions. Doing so, outer control loops can control high level features like path planning and obstacle avoidance without the need of being completely real-time. Thus, the UAV will keep position and wait for new commands by the high level layers.

This concept is working well in GPS conditions using the well tested out of the box algorithms of the new UAVs. In GPS denied environments, a position reference signal is necessary to implement the same concept. As all presented approaches have different advantages and disadvantages, the TRADR consortium is aiming for the following concept. A low level position control system is going to be implemented directly on the FCU using Stereo

Vision, Structured Light, Ultrasonic radar or a Lidar with very simple but robust algorithms. This system will only be locally stable without global knowledge and perception. But it will enable all higher level functions to take the necessary time for calculations and transmissions. In year 1, a lot of preliminary experiments were done to evaluate different sensors and algorithms. Furthermore this technology will be used to implement a low level obstacle avoidance system, the so called *virtual bumper*.

Having all these features together, high-level algorithms as well as teleoperators will be able to safely control the UAV in environments with and without access to GPS.

Corridor following One solution to automatic UAV flight in GPS denied environments can be achieved by finding and following the vanishing point of a corridor, using a light weight monocular camera (Figure 4). From the stream of images of the camera the diagonal lines are used to calculate the vanishing point. The UAV is then controlled to move towards this point autonomously. A basic solution can be found in [3] (Annex Overview 2.5).

Two forms of calculating the vanishing point were evaluated:

- based on edge detection principle
- based on density clustering of the pixels in a Sobel output image

In practice, the first method seems to be more robust and useful for simple and straight corridors. A drawback of both approaches is their sensitivity to changes in lighting conditions. None of the solutions include obstacle avoidance.

Capabilities of drone sensors In GPS denied environments, specific sensors have to be used to successfully be able to follow a waypoint path automatically. The measurement of relative distance to walls or obstacles is the most important problem in several of the real USAR scenarios.

In [21] (Annex Overview 2.3) and [20] (Annex Overview 2.2), monocular cameras, stereo cameras, 2D Laser, xTion, and 3D Laser were evaluated and compared concerning their capabilities to be used for obstacle detection.

Sensor	Points	Weight(g)	Size(cm)	Costs(€)
Mono	0	89	20x7x1	20
Stereo	3	202	40x7x1	40
2D Laser	4	370	6x6x9	4900
xTion	12	170	18x3x5	100
3D Laser	10	>370	>6x6x9	>4900

Table 1: Results of the sensor evaluation. The point score is based on the number of passed scenarios and the quality of the perceived environmental information.



Figure 5: UAV RGB image

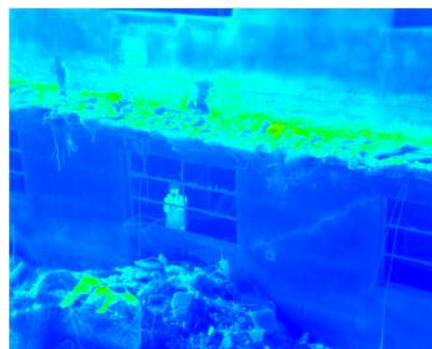


Figure 6: UAV infrared image

It turns out that monocular sensors are unsuitable for localization and for automatic navigation of UAVs, although they are lightweight and very cheap. They can estimate the relative motion of the sensor, but only under good conditions in terms of lighting and calibration. The position tracking is easily lost, and only small movements can be estimated robustly. If a feature-free method, such as optical flow is used, the estimation is indeed permanently given, but mostly unusable due to the integration of small estimation errors. Simple field experiments showed that it is possible to fly relative movements of 2-3 meters using PTAM (Parallel Tracking and Mapping Framework) [1]. However, rotations or other movements led to very poor estimates. In unstructured environments or structural conditions where strong rotations are necessary in order to move forward, this choice of sensor system is inappropriate.

The laser scanner delivers good results, but only in two dimensions. To get 3D laser scanner results, the sensor mount has to be actuated, which increases the total weight of the UAV. Still, if high quality maps are necessary, nevertheless the use of a rotating laser scanner can be preferred.

The xTion and the three-dimensional scanner passed every scenario and generated point clouds of the perceived environment. Based on this analysis, a reliable three-dimensional sensor as the xTion is proposed for automatic navigation of UAVs. Due to the low weight of the system it is easy to integrate, and the RGB camera can be useful in other tasks.

Further details about UAV-related path planning in 3D environments can be found in DR4.1.

TRADR Joint-Exercise experiences At the TJEx in Pisa, we used the *Pelican* and the *Falcon* UAV from Ascending Technologies [2]. A special feature of the later drone was the actively pan/tilt-balanced camera unit carrying infrared and high resolution RGB monocular cameras. By this mechanism the achieved images were much more stable and detailed and

thus more valuable to the firemen (Figure 5 and Figure 6).

Without the active mechanical stabilization the image stability and thus its quality was often reported as hard to work with. Furthermore the required change of the UAV batteries after about ten minutes was reported as too early. Making an extensive examination of the scenario ground without interruption very hard. All in all, the end-users appreciated the UAV as a very helpful tool to cope with rescue scenarios.

Future considerations The basic waypoint flight of the UAV should be enhanced by also controlling the yaw angle to keep the correct camera position on automatic flights.

For GPS denied environments, the approach based on vanishing point should run onboard the UAV. Thus, the UAV can navigate through corridors even without connection to an external computer. Also the robustness of the evaluation of the vanishing point detection should be increased.

Obstacle avoidance should run simultaneously together with waypoint or vanishing point flights. Also the integration of new sensors like the xTion should be pushed forward.

In the future the UAV should generally use a camera suspension to enhance image quality.

Concerning the operator control unit (OCU), efficient waypoint handling has to be integrated in order for the system to be usable by end-users.

1.3 Relation to the state-of-the-art

The teleoperation results are standard baseline solutions. They are not intended to go beyond the state of the art. This is however not the case for the UGV and UAV waypoint following functions.

Path planning and execution An overview of path planning techniques for mobile robotics can be found in [14] and in [13]. Purely 2D planning techniques are well studied and not applicable directly to the UGV and UAV as they operate in highly complex 3D disaster environments. This constraint also eliminates the use of a simplified 2.5D representation, a 2D grid augmented with altitude information as described in [17], as it cannot accurately describe building environment composed of several floors. We therefore have to cope with the high dimensionality of 3D path planning. The UGV waypoint following algorithm described in section 1.2.2 takes advantage of the fact that the robot operates over a 2D manifold in the 3D space, therefore simplifying the planning space, a strategy introduced by [16]. In every case, path planning is highly dependent of the environment representation quality and density. In order to achieve online capable solution, one needs to neglect re-meshing techniques, as considered by Wettergreen e.a. [19],

which are computationally very expensive. Within the UGV waypoint following framework, this representation densification is achieved by applying a tensor voting strategy described by Stumm e.a. [16]. This is one of many techniques used for digitally representing our 3D environment.

3D environment perception Research on the three-dimensional environment perception concerns the detection of structural features of the environment. Depending on the used sensor group, complex algorithms are necessary. Below, we will present current research approaches and classify them by their underlying detection manner. In particular, these results were considered during the evaluation of possible sensor groups for the future TRADR UAV.

Stereo visual methods are dealing with the detection of the environment based on two cameras. The method finds its origin in the biology and is based on the human vision. Papers by Lauterbach [11] and Klingbeil e.a. [9], address the integration of this perceptual process on the UAV. A recent approach of this kind is presented by Schmid et. al. (DLR) [15]. They used a stereo camera to get 3D data in the flight direction of a drone. The semi global block matching algorithm is implemented on a FPGA to reduce the huge computation demand of the stereo vision algorithm. Furthermore, first versions of 2D path planning and 3D mapping are also implemented on board. Real time performance is achieved with a patched Linux OS. The approach inherits typical stereo vision problems i.e. light dependencies, texturing of the environment and a limited field of view. Full 3D trajectory planning and dynamic obstacle avoidance are still open problems.

The mono visual reconstruction is concerned with the detection of environmental information based on continuous recordings of a single camera. There are several approaches that can be divided into the classes of point-based methods, or feature-based methods. The combination with high-quality sensor information, e.g., from a laser scanner, is also possible here [18], [9]. A new method which works neither on point-based nor on feature-based methods was presented by Engel e.a.[8] in 2013. The Parallel Tracking and Mapping Framework, (PTAM) [1], also provides a mono visual perception approach.

In addition to the visual environment perception, several sensors exist that can measure the distance to the obstacles directly. There are two-dimensional laser scanners, which are mounted on a rotating platform to assemble three dimensional scans which are stitched together using a method presented by Nüchter e.a.[12], to an overall model of the environment. Droschel e.a.[6] are also currently working on such a measurement method on the UAV. A novel sensor system is the Kinect by Microsoft or the Asus xTion. Both camera systems are projecting an infrared pattern and simultaneously measure the reflectance of this pattern. As a result, the sen-

sor produces three-dimensional measurements of the environment directly including RGB information. Endres et al. [7] present a method which registers those measurements based on their RGB information.

Recently, Intel also demonstrated a combination of structured light and stereo vision in their so called Real Sense cameras at minimum weight. These cameras were integrated to Ascending Technologies Firefly drones and demonstrated at Intel's CES Keynote January 6th 2015. The TRADR consortium will evaluate the capabilities of this technology as well. Doing so, TRADR will largely benefit from Ascending Technologies' access to this technology in close cooperation with Intel.

References

- [1] <http://www.robots.ox.ac.uk/~gk/PTAM/>.
- [2] <http://www.asctec.de/en/>.
- [3] Devvrat Arya. Autonomous UAV Navigation in Unknown Indoor Environment using Onboard Light Weight Camera. Technical report, Bonn-Rhein-Sieg University of Applied Sciences, January 2015.
- [4] Marius Beul, Rainer Worst, and Sven Behnke. Nonlinear Model-based Position Control for Quadrotor UAVs. In *Proceedings of Joint 45th International Symposium on Robotics (ISR) and 8th German Conference on Robotics (ROBOTIK)*, Munich, June 2014.
- [5] Francis Colas, Srivatsa Mahesh, François Pomerleau, Ming Liu, and Roland Siegwart. 3d path planning and execution for search and rescue ground robots. In *Intelligent Robots and Systems (IROS), 2013 IEEE/RSJ International Conference on*, 2013.
- [6] David Droschel, Jörg Stückler, and Sven Behnke. Local multiresolution representation for 6D motion estimation and mapping with a continuously rotating 3D laser scanner. In *Proc. of IEEE Int. Conf. on Robotics and Automation (ICRA)*, 2014.
- [7] F. Endres, J. Hess, N. Engelhard, J. Sturm, D. Cremers, and W. Burgard. An evaluation of the RGB-D SLAM system. In *Proc. of the IEEE Int. Conf. on Robotics and Automation (ICRA)*, St. Paul, MA, USA, May 2012.
- [8] J. Engel, J. Sturm, and D. Cremers. Semi-dense visual odometry for a monocular camera. In *Computer Vision (ICCV), 2013 IEEE International Conference on*, pages 1449–1456, Dec 2013.

- [9] Lasse Klingbeil, Matthias Nieuwenhuisen, Johannes Schneider, Christian Eling, David Droeschel, Dirk Holz, Thomas Läbe, Wolfgang Förstner, Sven Behnke, and Heiner Kuhlmann. Towards autonomous navigation of an uav-based mobile mapping system. In *4th International Conference on Machine Control & Guidance*, pages 136–147, 2014.
- [10] René Konze. Laserbasierte Hindernisvermeidung für autonome Kopertdrohnen. Bachelor’s thesis, Westphalian University of Applied Sciences Gelsenkirchen, July 2014. <http://publica.fraunhofer.de/dokumente/N-303821.html>.
- [11] Helge Andreas Lauterbach and Dipl-Ing Nils Gageik. Stereo-optische Abstandsmessung für einen autonomen Quadrocopter. 2013.
- [12] Andreas Nüchter, Kai Lingemann, Joachim Hertzberg, and Hartmut Surmann. 6D SLAM – 3D mapping outdoor environments. *J. Field Robot.*, 24(8-9):699–722, August 2007.
- [13] P Raja and S Pugazhenthii. Optimal path planning of mobile robots: A review. *International Journal of Physical Sciences*, 7(9):1314–1320, 2012.
- [14] N Sariff and Norlida Buniyamin. An overview of autonomous mobile robot path planning algorithms. In *Research and Development, 2006. SCOREd 2006. 4th Student Conference on*, pages 183–188. IEEE, 2006.
- [15] K. Schmid, P. Lutz, T. Tomić, E. Mair, and H. Hirschmüller. Autonomous vision-based micro air vehicle for indoor and outdoor navigation. *Journal of Field Robotics*, 31:537–570, 2014.
- [16] E Stumm, Andreas Breitenmoser, François Pomerleau, Cedric Pradalier, and Roland Siegwart. Tensor-voting-based navigation for robotic inspection of 3d surfaces using lidar point clouds. *The International Journal of Robotics Research*, 31(12):1465–1488, 2012.
- [17] Paul Tompkins, Anthony Stentz, and David Wettergreen. Global path planning for mars rover exploration. In *Aerospace Conference, 2004. Proceedings. 2004 IEEE*, volume 2, pages 801–815. IEEE, 2004.
- [18] Fei Wang, Jin-Qiang Cui, Ben-Mei Chen, and Tong H Lee. A comprehensive UAV indoor navigation system based on vision optical flow and laser fastSLAM. *Acta Automatica Sinica*, 39(11):1889 – 1899, 2013.
- [19] David Wettergreen, Scott Moreland, Krzysztof Skonieczny, Dominic Jonak, David Kohanbash, and James Teza. Design and field experimentation of a prototype lunar prospector. *The International Journal of Robotics Research*, 2010.

- [20] Stefan Wilkes. 3D Navigation eines UAVs in Umgebungen ohne GPS Empfang. Master's thesis, Westphalian University of Applied Sciences Gelsenkirchen, January 2015. <http://publica.fraunhofer.de/dokumente/N-???????.html>.
- [21] Stefan Wilkes. 3D Navigation for UAVs in GPS denied environments. Unpublished draft paper, January 2015.
- [22] Karel Zimmermann, Petr Zuzanek, Michal Reinstein, and Vaclav Hlavac. Adaptive traversability of unknown complex terrain with obstacles for mobile robots. In *Robotics and Automation (ICRA), 2014 IEEE International Conference on*, pages 5177–5182. IEEE, 2014.

2 Annexes

In this section we will list the papers included in this deliverable. For each paper we list authors, title, abstract and a short description how this relates to the work described above.

2.1 Beul et al. (2014), “Nonlinear Model-based Position Control for Quadrotor UAVs”

Bibliography Marius Beul, Rainer Worst, Sven Behnke. “Nonlinear Model-based Position Control for Quadrotor UAVs.” In *Proceedings of Joint 45th International Symposium on Robotics (ISR) and 8th German Conference on Robotics (ROBOTIK)*. Munich, June 2014.

Abstract Position control of Micro Air Vehicles (MAV) is challenging, because position measurements by global navigation satellite systems or laser scanners are typically available at much lower rates than the control frequency. Furthermore, the transient response of classic PID controllers is either slow or induces overshoot. In this work, we address this issue by a model-based control approach. We model and identify the dynamics of the MAV and use this knowledge in a nonlinear cascaded controller to generate time-optimal trajectories. The proposed method is evaluated in simulation and two real MAVs.

Relation to WP This contribution describes a model-based controller for the NIFTi UAV, which can be used in the context of T2.4.

2.2 Wilkes (2015), “3D Navigation for UAVs in GPS denied environments (master’s thesis)”

Bibliography Stefan Wilkes. “3D Navigation for UAVs in GPS denied environments.” Master’s thesis, Westphalian University of Applied Sciences Gelsenkirchen, January 2015.

Abstract The master thesis “3D Navigation for UAVs in GPS denied environments” is concerned with the automatic navigation of an unmanned aerial vehicle in three dimensional space. It analyses and explains several sensor groups and their corresponding algorithms to perceive the environment in urban search and rescue missions. Afterwards the results are going to be evaluated. Furthermore a first approach for the implementation of such a navigation is presented, which can be used both in simulation and on an arbitrary UAV, due to the Robot Operating System.

Relation to WP Within this thesis, a method for 3D navigation has been developed, which was used in the context of T2.4.

2.3 Wilkes (2015), “3D Navigation for UAVs in GPS denied environments (draft paper)”

Bibliography Stefan Wilkes. “3D Navigation for UAVs in GPS denied environments.” Unpublished draft paper, January 2015.

Abstract The master thesis “3D Navigation for UAVs in GPS denied environments” is concerned with the automatic navigation of an unmanned aerial vehicle in three dimensional space. It analyses and explains several sensor groups and their corresponding algorithms to perceive the environment in urban search and rescue missions. Afterwards the results are going to be evaluated. Furthermore a first approach for the implementation of such a navigation is presented, which can be used both in simulation and on an arbitrary UAV, due to the Robot Operating System. This paper presents a short summary and the most important results of the entire master thesis, which is written in German.

Relation to WP This draft paper summarizes the results of a Master’s thesis[20] (Annex Overview 2.2) in the context of T2.4.

2.4 Konze (2014), “Laser-based obstacle avoidance for autonomous multicopters”

Bibliography René Konze. “Laser-based obstacle avoidance for autonomous multicopters.” Bachelor’s thesis, Westphalian University of Applied Sciences Gelsenkirchen, July 2014.

Abstract In this thesis, a method was developed and implemented that enables the TRADR UAV to detect and avoid obstacles during its automatic GPS-based waypoint following.

Relation to WP Within this thesis, a method for obstacle detection has been developed, which was used in the context of T2.4.

2.5 Arya (2015), “Autonomous UAV Navigation in Unknown Indoor Environment using Onboard Light Weight Camera.”

Bibliography Devvrat Arya. “Autonomous UAV Navigation in Unknown Indoor Environment using Onboard Light Weight Camera.” Unpublished report on R&D, January 2015.

Abstract The field of aerial vehicles is nowadays a popular and fast evolving one. In recent years, robotics community has shown an increasing interest in autonomous flying robots as they can well implement many flight missions in more challenging environments, with lower risk of damaging itself and its surroundings. Miniature aerial vehicles (MAVs) that can fly autonomously in unstructured indoor GPS-denied environments such as homes and offices would be useful for exploration, rescue, surveillance and also for entertainment. These aerial vehicles need to be small and light weight which restricts them with limited payload and power, so a low power consuming long range sensor is required instead of power consuming LIDARs or Lasers. In this report we propose a system that enables a low cost quadcopter coupled with a ground based laptop to fly autonomously in a previously unknown and GPS denied environments such as indoor corridors or hallways. The proposed method is simple and based on onboard robust monocular camera. The aim is to find the vanishing point by using least computationally expensive image processing technique and make the drone fly towards the vanishing point. We also discussed the difficulties in achieving fully autonomous UAV flight, highlighting the differences between ground and aerial robots that make it difficult to use algorithms that have been developed for ground robots. The primary contribution is the development of a basic indoor navigation system for UAV. We also show experimental results that validates the UAV's ability to fly in unknown corridors, and demonstrate the robustness of the algorithm implemented.

Relation to WP This report describes an approach for autonomous flight based on the detection of vanishing points in the context of T2.4.

Nonlinear Model-based Position Control for Quadrotor UAVs

Marius Beul, University of Bonn, mbeul@ais.uni-bonn.de, Germany

Rainer Worst, Fraunhofer IAIS, rainer.worst@iais.fraunhofer.de, Germany

Sven Behnke, University of Bonn, behnke@cs.uni-bonn.de, Germany

Abstract

Position control of Micro Air Vehicles (MAV) is challenging, because position measurements by global navigation satellite systems or laser scanners are typically available at much lower rates than the control frequency. Furthermore, the transient response of classic PID controllers is either slow or induces overshoot.

In this work, we address this issue by a model-based control approach. We model and identify the dynamics of the MAV and use this knowledge in a nonlinear cascaded controller to generate time-optimal trajectories. The proposed method is evaluated in simulation and two real MAVs.

Keywords: Position control, Model based control, MAV, UAV, Quadrotor

1 Introduction

In recent years, micro aerial vehicles (MAVs) have become widely available. Due to their low cost and flexibility, they are used for aerial photography, inspection, surveillance and rescue missions.

In most cases, a human operator pilots the MAV remotely to fulfill a specific task or the MAV is following a predefined path of GPS waypoints in an obstacle-free altitude. Instead of remotely operating the MAV, we aim for a fully autonomous flight.

For the above mentioned tasks, a high level of autonomy is necessary, including the capability of flying to and staying at waypoints. To this end, a model-based position controller is developed in this work. Particular attention is needed in terms of overshoot and settling time of the controller. During missions in restricted environments such as urban areas with close-to-wall flying, overshoot could easily lead to collisions. Time is also a crucial asset in these operations, since the battery strictly limits the achievable flight time. Section 2 briefly describes the MAVs used in this work.

2 Micro Aerial Vehicles

2.1 MAV 1

Successful execution of rescue operations demand quick response from the fire-fighters which may cause physical and psychological stress on them during emergency services. In order to facilitate them to perform their task efficiently, a MAV (**Fig. 1**) is developed to support such operations. The MAV serves as a mobile sensor platform and operates in coopera-

tion with the humans involved. For a comprehensive specification of the MAV properties see [1]. The setup can be summarized as follows:

Sensors:

- 1× 2D Laser scanner
- 1× GPS
- 1× Inertial measurement unit (IMU)
- 1× Camera dome

Processing:

- 1× Intel-Atom 1.6 GHz
- 1× Mikrokopter FlightCtrl

Actuators:

- 8× Coaxial Robbe ROXXY 2827-35



Figure 1: MAV with GPS, laser scanner and camera dome used in rescue operations

2.2 MAV 2

Furthermore, we create mission-specific semantic maps on demand. Special focus lies on the inspection of a building's facade [2]. Hence, the MAV has to operate in the vicinity of buildings and other structures, e.g. trees and power cables. For this purpose, a planning algorithm generates optimal paths through the previously mapped environment (**Fig. 2**). Further information on planning is found in [3].

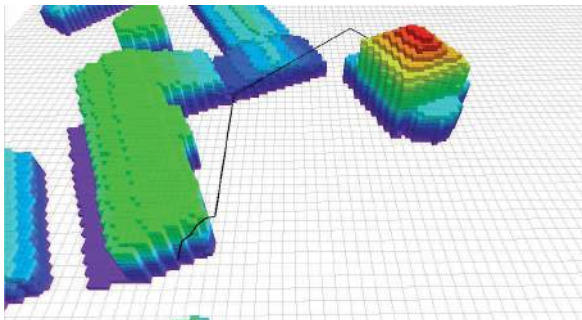


Figure 2: Map of the surroundings of the MAV with planned trajectory

Our MAV used for these tasks is shown in **Fig. 3**. It is equipped with a variety of sensors with complimentary properties.



Figure 3: MAV equipped with DGPS, stereo cameras and 3D laser scanner

For a detailed description of our sensor setup and the processing pipeline see [4, 5, 6]. The setup can be summarized as follows:

Sensors:

- 2× Fisheye stereo cameras
- 1× 3D laser scanner (rotating 2D scanner)
- 1× Motion camera [7]
- 1× Differential GPS (DGPS) [8]
- 1× Inertial measurement unit (IMU)
- 8× Ultrasonic distance sensors

Processing:

- 1× Intel Core i7 3820QM 2.7GHz
- 1× Pixhawk Autopilot

Actuators:

- 8× Coaxial MK3638 Motors

3 Related Work

Most traditional position controllers are based on standard proportional-integral-derivative (PID) controllers. Commercially available platforms like the Mikrokopter, the PX4 or the OpenPilot CopterControl use linear PID-controllers for positioning.

Li et al. [9] and R. Baránek et al. [10] create a dynamic model of a quadrotor. Positioning is also achieved with classic PID-control based on parameters obtained from simulation. Puls et al. [11] describe a PI-controlled quadrotor. It relies on a dynamic model and is enhanced with a correction term to lead the quadrotor on a straight path to the target. A linear state-space model is identified and parameterized by Pfeifer et al. [12]. Subsequently, a linear state-space controller is implemented and parameterized via pole placement. Bouabdallah et al. [13] derive a model from differential equations. Basic PID and backstepping control techniques are combined to control attitude, height, and position of the quadrotor. A nonlinear model of a MAV is created by Patel et al. [14]. It consists of a linear and a nonlinear part which are controlled separately by PID and sliding mode control. Some works employ machine learning techniques for quadrotor control. Dierks et al. [15], for example, use neural networks to learn the quadrotor dynamics and for positioning. All approaches have in common, that multiple parameters and gains have to be adjusted. Either simple PID gains or complex model parameters have to be found to achieve a good transient response. In this work, a model with very few physically meaningful parameters is derived, which is identified and used for model-based control.

4 Modeling MAV Dynamics

4.1 Physics-based Model

A grey-box model of the 2D-dynamics of the MAV is developed. It is assumed that the MAV is symmetrical

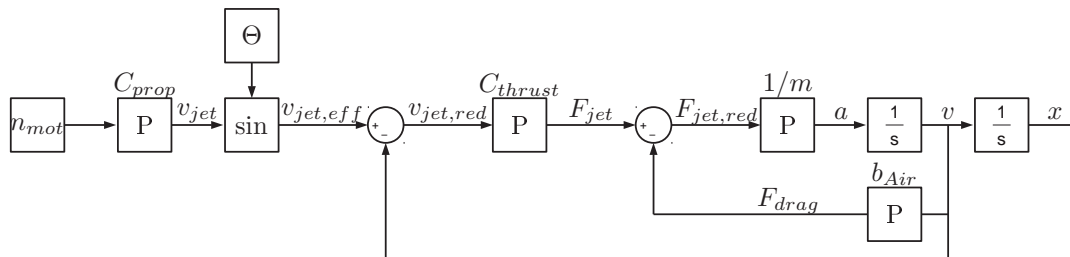


Figure 4: Grey-box model of the 1D-kinematics of the MAV with two DoF.

and thus can be modelled as a superposition of two identical models with two degrees of freedom (DoF) each (deflection Θ and position x). It is also assumed that the MAV is equipped with an underlying attitude and an overlying height controller. Considering differential equations of motion, **Fig. 4** shows an approach for the model. The MAV is modeled as point mass with state variables $[v, x]$ (velocity and position).

Assuming the MAV is hovering at constant height, the rotation speed of all motors n_{mot} results in a constant jet stream v_{jet} . This is represented by constant C_{prop} , which depends on aerodynamic properties of the propellers. The direction of the jet stream is governed by the deflection of plant input Θ . The resulting jet $v_{\text{jet,eff}}$ is reduced by the movement of the MAV v and amplified by the thrust constant C_{thrust} . This constant represents the size of the jet stream and the aerodynamic properties of the MAV. Reduced by the drag and concerning the mass of the MAV, this force propels the MAV with acceleration a which results in the velocity v and furthermore in the movement x of the MAV.

With the following restrictions made, the model can be massively simplified to the double integrator shown in **Fig. 5**.

- Small angular deflection ($\sin(v_{\text{jet}}) \approx v_{\text{jet}}$)
- Slow horizontal movement ($v_{\text{jet,eff}} \gg v$)
- Constant height ($n_{\text{mot}} \approx \text{const.}$)
- Negligibly small drag ($F_{\text{jet}} \gg F_{\text{drag}}$)

The plant input Θ is amplified with the model specific gain C_{acc} , which results in the acceleration a of the MAV that is integrated to the velocity v and the movement x .

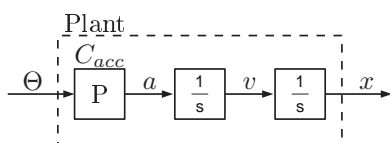


Figure 5: Simplified grey-box model of the 1D-kinematics of the MAV with two DoF.

4.2 Parameter Identification

The model is fitted with experimental data. For this model only one parameter has to be identified. This is done by fitting the MAV model with experimental data obtained in various test flights using gradient descend. For our MAVs, equations 1 to 3 hold;

$$C_{\text{acc}} = \frac{C_{\text{prop}} C_{\text{thrust}} n_{\text{motor}}}{m}, \quad (1)$$

$$C_{\text{acc, MAV1}} = 9.3 \frac{\text{m}}{\text{s}^2}, \quad (2)$$

$$C_{\text{acc, MAV2}} = 8.5 \frac{\text{m}}{\text{s}^2}. \quad (3)$$

5 Model-based Control

Based on the identified model, a nonlinear controller is developed (**Fig. 6**). We limit the allowed deflection of the MAV in order to avoid high-speed or dynamic flight maneuvers, which could be dangerous in the vicinity of obstacles. Large deflections would also prevent the linearization of the model in Fig. 5. Despite these precautions, overshoot is not permissible as it could lead to collisions during close-to-wall flying.

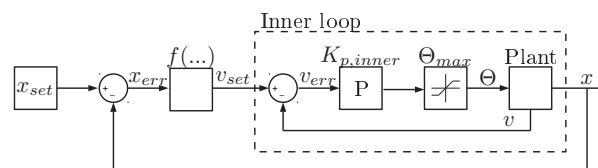


Figure 6: Closed loop control structure.

It can be seen that a cascaded control loop is used to control the position x as well as the velocity v of the MAV.

The inner loop consists of a P-controller which set-point is driven by an outer loop. Although the outer loop could also be a P-controller to archive perfect transient responses (infinitely small settling time without overshoot) in a non-limited system, here the outer loop has to be nonlinear.

Considering simple equations of motion, Eq. 4 shows

the nonlinear part of the controller $f(\dots)$. With respect to the limited plant input, this controller is capable of achieving time-optimal responses without the handicap of adjusting multiple gains:

$$f(\dots) = \sqrt{2 \frac{\Theta_{max} \cdot C_{acc}}{x_{err}}} \quad (4)$$

Since both axes are controlled separately, the resulting trajectory to the target is bowed. This behavior is shown in **Fig. 7**.

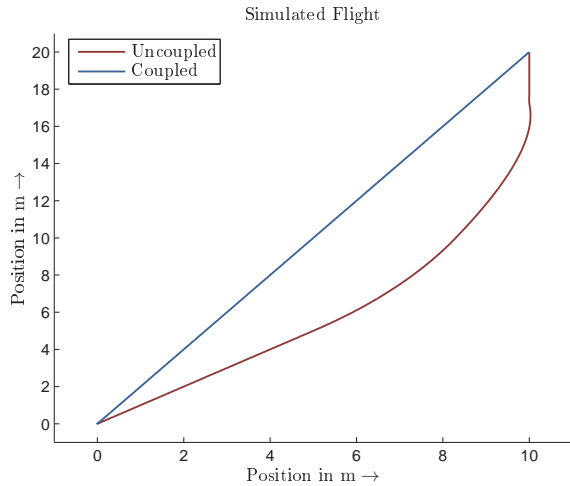


Figure 7: Flight trajectory with decoupled and coupled axes ($x_{start} = [0 \text{ m}, 0 \text{ m}]$, $x_{stop} = [10 \text{ m}, 20 \text{ m}]$, $\Theta_{max} = 5^\circ$, $C_{acc} = 9.3 \frac{\text{m}}{\text{s}^2}$, $K_{p,inner} = 10 \frac{\text{s}}{\text{m}}$).

This issue is addressed by defining a master, and a slave axis, following the idea proposed for example in [16]. Both axes predict the time of arrival on the next waypoint from the current state. This is done analytically by solving Eq. 5 to Eq. 8.

$$\int v dt = x_{err} \quad (5)$$

$$\int v dt = vt_{acc} + \frac{1}{2}v_{max}t_{dec} + \frac{1}{2}(v_{max} - v)t_{acc} \quad (6)$$

$$v_{max} = v + t_{acc}\Theta_{max}C_{acc} \quad (7)$$

$$t = t_{acc} + t_{dec} \quad (8)$$

The solution is

$$t = -\frac{v}{\Theta_{max}C_{acc}} + \sqrt{\frac{v^2}{2 \cdot (\Theta_{max}C_{acc})^2} + \frac{x_{err}}{\Theta_{max}C_{acc}}}$$

The master is defined as the axis with the higher time of arrival. Subsequently Θ_{max} of the slave axis is set to match the arrival time:

$$\Theta_{max,sl} = \frac{-\frac{v_{sl}}{t_{ma}} + \frac{2x_{sl}}{t_{ma}^2} + \sqrt{(\frac{v_{sl}}{t_{ma}} - \frac{2x_{sl}}{t_{ma}^2})^2 + \frac{v_{sl}^2}{t_{ma}^2}}}{C_{acc}}$$

6 Experiments

The algorithm is first implemented and evaluated in simulation (Section 6.1). In Section 6.2, it is applied and evaluated on the MAV.

6.1 Simulation

Fig. 8 shows the simulated step response and subsequent position hold.

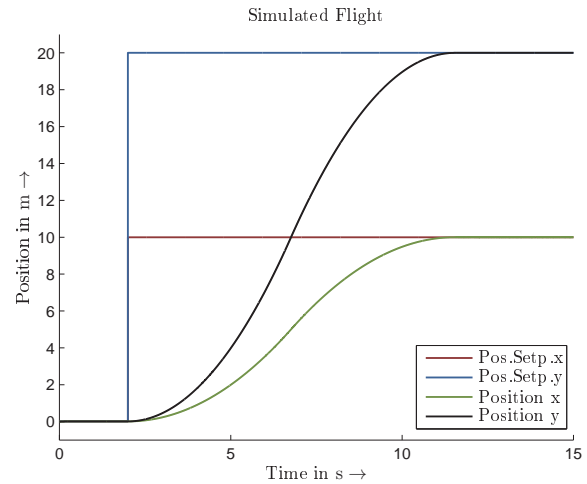


Figure 8: Step response in simulation ($x_{start} = [0 \text{ m}, 0 \text{ m}]$, $x_{stop} = [10 \text{ m}, 20 \text{ m}]$, $\Theta_{max} = 5^\circ$, $C_{acc} = 9.3 \frac{\text{m}}{\text{s}^2}$, $K_{p,inner} = 10 \frac{\text{s}}{\text{m}}$).

The feedback for the controller in simulation contains no noise and has an update rate of 1 kHz. The coupled behavior is also shown in Fig. 7. **Fig. 9** shows the corresponding velocity trajectories.

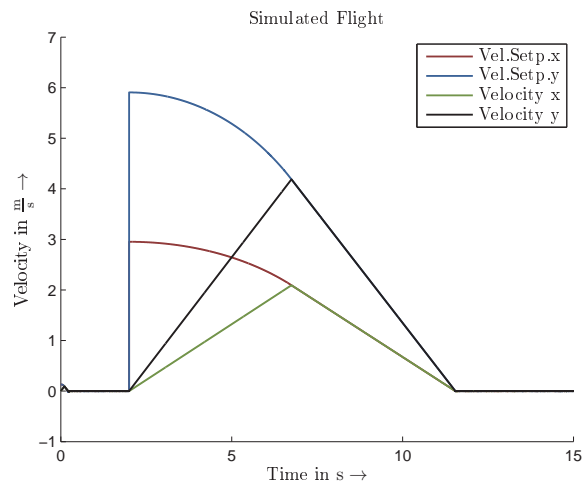


Figure 9: Step response in simulation ($x_{start} = [0 \text{ m}, 0 \text{ m}]$, $x_{stop} = [10 \text{ m}, 20 \text{ m}]$, $\Theta_{max} = 5^\circ$, $C_{acc} = 9.3 \frac{\text{m}}{\text{s}^2}$, $K_{p,inner} = 10 \frac{\text{s}}{\text{m}}$).

As can be seen in Fig. 8, these profiles lead to exact positioning in both axes at the same time. By limiting

the deflection of the slave axis, an unbowed yet time optimal trajectory is generated.

6.2 Real MAV Flight

The control algorithm is also evaluated in real MAV-flight. **Fig. 10** shows a transient response, recorded with MAV 1.

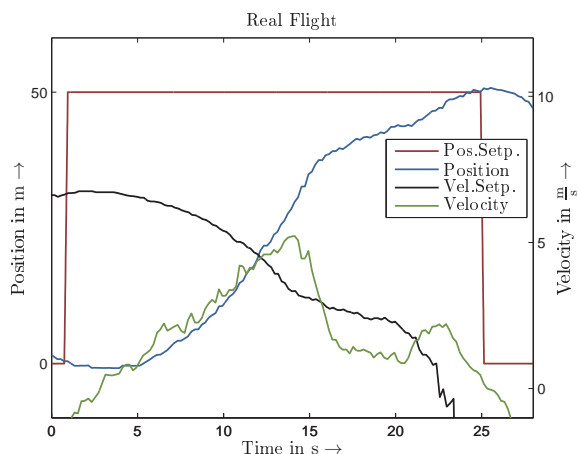


Figure 10: Step response ($x_{start} = 0$ m, $x_{stop} = 50$ m, $\Theta_{max} = 3^\circ$, $C_{acc} = 9.3 \frac{m}{s^2}$, $K_{p,inner} = 10 \frac{s}{m}$).

It can be seen that the feedback provided by the on-board GPS is much less accurate than the simulated feedback. It leads to overshoot in the velocity. Nevertheless, the overshoot in the position is negligibly small. Furthermore, it can be seen that for example at $t = 12$ s – 16 s, the measured velocity decreases faster than the planned velocity profile. This is an indication for modelling uncertainties.

The algorithm is compared to the existing Mikrokopter position controller. **Table 1** shows the results.

Controller	Θ_{max}	Settling Time	Overshoot
Nonlinear	2°	45 s	1.3 m
Nonlinear	3°	18 s	0.7 m
Nonlinear	5°	14 s	3.3 m
Mikrokopter	-	18 s	2.2 m

Table 1: Performance of the controller

It can be seen that even with bad feedback (no DGPS available) and error-prone parameterization the nonlinear approach shows better results than the original controller. For this very feedback (GPS with 5 Hz) and model parameterization, a maximum deflection of $\Theta_{max} = 3^\circ$ would be recommended.

7 Conclusions

In this paper, an approach for a model-based position controller for an unmanned MAV was presented.

A simplified model is derived from differential equations. Model parameters are fitted to the real system to approximate the MAV dynamics. A nonlinear cascaded controller, which is capable of handling the strictly limited plant input, is proposed. The control algorithm is implemented in simulation and on a real MAV. It is evaluated and compared to the existing system.

Due to the easy model identification process and the ability to reach waypoints without overshoot, the approach proposed in this paper is applicable to close-to-wall flying. The ability to stay on a linear trajectory in combination with the fast transient response make the controller ideal for MAVs with limited deflection. Since the dynamics is limited by the slow and inaccurate feedback, additional research will address this issue. Especially the use of DGPS on MAV 2 will be subject to further research. Also the ability to pass waypoints at a certain speed will be investigated. Furthermore, including height control as a third coupled axis will lead to straight paths in 3D space.

8 Acknowledgments

This research is supported by the EU FP7 ICT , Project #247870FP7 NIFTi and grant BE 2556/8 of German Research Foundation (DFG).

References

- [1] V. Tretjakov, J. Winzer and R. Worst. Platform manufacturing and sensor integration (UAV). *Fraunhofer IAIS*, 2010
- [2] S. Loch-Dehbi, Y. Dehbi and L. Plümer. Stochastic reasoning for UAV supported reconstruction of 3D building models. *ISPRS XL-1/W2*, 2013
- [3] M. Nieuwenhuisen and S. Behnke. Hierarchical Planning with 3D Local Multiresolution Obstacle Avoidance for Micro Aerial Vehicles. *ISR / Robotik*, 2014
- [4] M. Nieuwenhuisen, D. Droschel, J. Schneider, D. Holz, T. Läche, and S. Behnke. Multimodal obstacle detection and collision avoidance for micro aerial vehicles. *6th European Conference on Mobile Robots (ECMR)*, 2013
- [5] D. Droschel, J. Stückler, and S. Behnke. Local multiresolution representation for 6D motion estimation and mapping with a continuously rotating 3D laser scanner. *IEEE International Conference on Robotics and Automation (ICRA)*, 2014

- [6] L. Klingbeil, M. Nieuwenhuisen, J. Schneider, C. Eling, D. Droschel, D. Holz, T. Läbe, W. Förstner, S. Behnke and H. Kuhlmann. Towards Autonomous Navigation of an UAV-based Mobile Mapping System. *International Conference on Machine Control & Guidance*, 2014
- [7] D. Honegger, L. Meier, P. Tanskanen and M. Pollefeys. An open source and open hardware embedded metric optical flow CMOS camera for indoor and outdoor applications. *IEEE International Conference on Robotics and Automation (ICRA)*, 2013
- [8] C. Eling, L. Klingbeil, M. Wieland and H. Kuhlmann. A precise position and attitude determination system for lightweight unmanned aerial vehicles. *ISPRS*, 2013.
- [9] J. Li and Y. Li. Dynamic analysis and PID control for a quadrotor. *Proc. of IEEE International Conference on Mechatronics and Automation*, 2011
- [10] R. Baránek and F. Šolc. Modelling and control of a hexa-copter. *13th International Carpathian Control Conference (ICCC)*, 2012
- [11] T. Puls, M. Kemper, R. Küke and A. Hein. GPS-based position control and waypoint navigation system for quadcopters. *International Conference on Intelligent Robots and Systems*, 2009
- [12] E. Pfeifer, F. Kassab Jr. Dynamic feedback controller of an unmanned aerial vehicle. *Brazilian Robotics Symposium and Latin American Robotics Symposium*, 2012
- [13] S. Bouabdallah and R. Siegwart. Full control of a quadrotor. *Proceedings of the IEEE/RSJ IROS*, 2007
- [14] A. R. Patel, M. A. Patel and D. R. Vyas. Modeling and analysis of quadrotor using sliding mode control. *44th IEEE Southeastern Symposium on System Theory*, 2012
- [15] T. Dierks and S. Jagannathan. Output feedback control of a quadrotor UAV using neural networks. *IEEE Transactions on Neural Networks*, 2010.
- [16] D. A. Pierre. Optimization Theory with Applications. *Dover Publications*, 1986



**Westfälische
Hochschule**

Gelsenkirchen Bocholt Recklinghausen



Fraunhofer

IAIS

Masterthesis

**3D Navigation for UAVs in GPS denied
environments**

3D Navigation eines UAVs in Umgebungen ohne GPS Empfang

Vorgelegt von: Stefan Wilkes
Hartmannsweilerstr. 16a
46119 Oberhausen
200823277

Datum: 12. Januar 2015

Gutachter: Prof. Dr. Hartmut Surmann
Dipl. Inf. Rainer Worst

Datum und Unterschrift des Betreuers

Inhaltsverzeichnis

1	Einleitung	1
1.1	Motivation	1
1.2	Problemstellung	3
1.2.1	Kernproblematik	4
1.2.2	Teilproblematiken	4
1.3	Struktur der Arbeit	4
2	Stand der Technik	5
2.1	Unmanned Aerial Vehicles	5
2.2	Dreidimensionale Umgebungswahrnehmung	6
2.3	Robot Operating System	10
3	Systemdesign	11
3.1	Software	11
3.2	Hardware	12
3.2.1	Ascending Technologies: Pelican	12
3.2.2	Parrot ARDrone	15
3.2.3	Stereo RGB-D Testsystem	16
3.2.4	3D Laserscanner	16
3.3	Inertial Measurement Unit (IMU)	17
3.3.1	Bestimmung der Orientierung durch Beschleunigungssensor und Magnetometer	18
3.3.2	Bestimmung der Orientierung durch Gyroskop	19
3.3.3	Optimierung der Orientierung durch Fusion der beiden Verfahren	20
3.4	Struktur des Gesamtsystems	20
4	Verfahren zur Wahrnehmung der Umgebung	22
4.1	Mono visuelle Odometrie und Strukturen durch Bewegung	22
4.1.1	Grundlagen	22
4.1.2	Bestimmung der intrinsischen Parameter	24
4.1.3	Die Epipolareometrie	28
4.1.4	Detektieren korrespondierender Punkte	32
4.1.5	Berechnung der visuellen Odometrie	39
4.1.6	Triangulation der dreidimensionalen Umgebung	40
4.1.7	Das Skalierungsproblem	42
4.1.8	Implementierung des Frameworks	43
4.2	Parallel Tracking and Mapping (PTAM)	44
4.3	Stereo Vision	45
4.3.1	Bestimmung der extrinsischen Parameter	46
4.3.2	Rektifizierung der Eingangsdaten	48
4.3.3	Rekonstruktion der Tiefeninformation	49
4.3.4	Rekonstruktion der dreidimensionalen Umgebung	52

4.4	Tiefensenorik	53
4.4.1	RGB-D Sensor	53
4.4.2	Laserscanner	54
4.5	3D Scanner	55
5	Evaluierung der Verfahren	57
5.1	Definition geeigneter Testdatensätze	57
5.2	Distanz- und Genauigkeitsmessungen	64
5.3	Ergebnis der Evaluierung	65
6	Navigation	69
6.1	Dreidimensionale Pfadplanung	69
6.1.1	C-Space Diskretisierung	70
6.1.2	Globale Pfadplanung	72
6.1.3	Path tracking	76
6.1.4	Lokale Navigation	77
6.1.5	MoveIt! Framework	79
6.2	Simulationsumgebung	81
6.2.1	ICP Algorithmus	82
7	Abschlussbetrachtung	84
7.1	Zusammenfassung	84
7.2	Ausblick	85
8	Anhang	87
8.1	Planung der Arbeit	87
8.2	Aufwandsanalyse	90
8.2.1	Quellcode	90
8.2.2	Aufgebrachte Stunden	91
8.3	Vergleich: Sequentielle zu parallele Implementierung	92

Abbildungsverzeichnis

1	Roboter der Firma iRobot erkunden das Atomkraftwerk in Fukushima und liefern Videomaterial des Geschehens (Foto: Tepco)	1
2	Ruine eines ehemaligen Militärhospiz in Pisa. Testgelände für das TRADR Projekt. Viele der Trümmerteile sind auch für flexible Bodenroboter nur schwer oder gar nicht passierbar.	2
3	Kommerzieller Nano Quadroter[40]. Ausgestattet mit grundlegender Stabilitätsensorik. Gewicht ca. 300g. Kosten: Ca. 50 Euro.	5
4	Kommerzieller semi-professioneller Quadroter DJI Phantom II[47]. Ausgestattet mit verschiedenen Sensoren zur Positions- und Inertialsmessung. Ebenfalls mit Kamerasystem und GPS ausgestattet. Kosten: Ab 599 Euro.	6