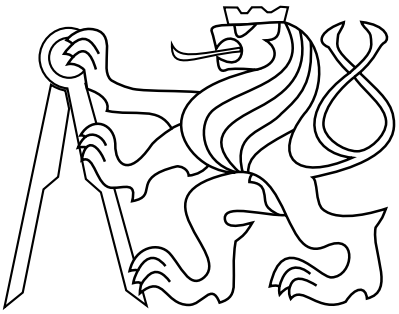
Czech Technical University in Prague

Faculty of Electrical Engineering
Department of Cybernetics
Multi-robot Systems



A Complex System for Fast and Reliable Deployment of Cooperating Autonomous Aerial Vehicles

Doctoral Thesis

Ing. Vojtěch Spurný

Prague, August 2021

Ph.D. Programme: Electrical Engineering and Information Technology
Branch of Study: Artificial Intelligence and Biocybernetics

Supervisor: doc. Ing. Martin Saska, Dr. rer. nat.

Acknowledgments

I would like to thank my supervisor, Martin Saska, for his great guidance during my studies and research. Furthermore, I would like to thank all my colleagues from Multi-robot systems (MRS) group for their friendship and help throughout my studies. I thank Tomáš Báča, Viktor Walter, Matěj Petrlík, Matouš Vrba, Daniel Heřt, Vít Krátký, Pavel Petráček, Petr Štibinger, and many others. Without their enormous work on the MRS system for UAVs, it would not be possible to accomplish the results presented in this thesis. Finally, special thanks go to my friends and my family for their support. Above all, I am extremely grateful to my Zuzana for the final push to finish my studies.

During my PhD studies, I have been supported by Czech Republic through PhD scholarship. The Czech Science Foundation supported this work through projects No. 17-16900Y, No. 19-22555Y, No. 20-10280S, and No. 20-29531S. The Czech Technical University in Prague supported my work with grants SGS15/157/OHK3/2T/13, SGS17/187/OHK3/3T/13, and SGS20/174/OHK3/3T/13. The Ministry of Education of the Czech Republic has funded my research by OP VVV funded project CZ.02.1.01/0.0/0.0/16_019/0000765 "Research Center for Informatics". The Ministry of Culture of the Czech Republic has funded my research by R&D project No. DG18P02OVV069 from programme NAKI II. European Union supported my work through H2020 project AERIAL CORE, No. 871479. The Khalifa University of Science funded our participation in the MBZIRC 2017 and MBZIRC 2020 competitions that also motivated this work.

Copyright

Abstract

In time-sensitive scenarios such as disaster response, first responders face increased technical challenges. These challenges may include rugged terrain, unstable structures, degraded environmental conditions, severe communication constraints, and extensive operation areas. To ensure the safety of human responders, autonomous robots such as Unmanned Aerial Vehicles (UAVs) are well suited for this task. At the beginning of the research for this thesis, the testing standard in the UAV community was to use a simulated environment or to conduct short isolated experiments under laboratory conditions relying on a motion capture system with no external disturbances and a stable communication infrastructure without interference. Compared to these approaches, real-world deployment raises additional constraints and challenges to fundamental research problems. A robotic system for a real-world application requires a reliable high-level planning architecture to recover from robot failures and malfunctions of its system parts. This is essential for the UAV and especially for the multi-UAV systems that are the focus of this thesis. Communication between the robots is required to fully access the capabilities of the deployed multi-robot team. As of today however, the technical requirements for an operational communication infrastructure are still a bottleneck. Therefore, the first part of the thesis is dedicated to developing a novel high-level planning architecture that considers different strategies based on the availability of a communication infrastructure in addition to failures of sensors and actuators. When enabled, this maximizes the contributions and tight cooperation of the multi-robot system. The second part of the thesis focuses on motion planning for multiple cooperating UAVs inspired by three real-world scenarios: autonomous delivery of objects by a team of UAVs, autonomous aerial surveys of building interiors, and autonomous firefighting. The object delivery and firefighting scenarios were motivated by challenges in the Mohamed Bin Zayed International Robotics Challenge (MBZIRC) 2017 and 2020.

Keywords: Unmanned Aerial Vehicle, Autonomous Systems, Mobile Robots, Motion Planning

Abstrakt

V situacích, kdy je nutné reagovat okamžitě, jako například při různých katastrofách, čelí záchranáři zvýšeným technickým výzvám. Často musí operovat v rozsáhlých oblastech, vzdorovat nestabilnímu terénu a obtížným podmínkám prostředí, kde mají omezené komunikační možnosti. K usnadnění jejich práce mohou velmi efektivně sloužit autonomní roboti, mezi které se řadí například bezpilotní helikoptéry. Na začátku výzkumu pro tuto práci bylo v robotické komunitě obvyklé používat simulované prostředí nebo krátké izolované experimenty v laboratorním podmínkách. V porovnání s uvedenými prostředími přináší reálné nasazení mnohá omezení. Autonomní robotický systém v reálných podmínkách vyžaduje spolehlivou rozhodovací architekturu, která dokáže adekvátně reagovat na selhání nebo výpadek části nebo částí systému. Tato schopnost je zásadní pro bezpilotní helikoptéry a zejména pak pro multi-robotické týmy skládající se z několika bezpilotních helikoptér, které jsou tématem této práce. Pro maximální využití potenciálu multi-robotického týmu je zapotřebí mít spolehlivou komunikační infrastrukturu, která však ani v dnešní době není snadno dosažitelná. První část této práce je zaměřena na vytvoření rozhodovací architektury, která dokáže reagovat nejen na chyby senzorů a aktivních prvků, ale také volit nejvhodnější strategii v závislosti na dostupnosti komunikace. To umožňuje maximalizovat efektivitu multi-robotického týmu a jejich blízkou spolupráci, kdykoli je to možné. Druhá část práce je zaměřena na plánování spolupráce autonomních bezpilotních helikoptér inspirované třemi reálnými scénáři, které jsou: autonomní doručení objektů, autonomní bezpečný průzkum interiérů budov a autonomní hašení požárů. Doručení objektů a hašení požárů bylo motivováno úkoly v mezinárodní robotické soutěži MBZIRC konané v roce 2017 a 2020.

Klíčová slova: Bezpilotní helikoptéry, Autonomní systémy, Mobilní roboti, Plánování pohybu

Contents

Abbreviations **vii**

1 Introduction **1**
1.1 Thesis outline 5

2 Contributions **6**
2.1 High-level planning architecture 6
2.2 Motion planning for multiple UAVs 8
2.3 System for facilitation of transfer from fundamental achievements to real experiments 11

3 Autonomous Firefighting Inside Buildings by an Unmanned Aerial Vehicle **13**

4 Localization, Grasping, and Transportation of Magnetic Objects by a team of MAVs in Challenging Desert-like Environments **33**

5 Cooperative Autonomous Search, Grasping, and Delivering in a Treasure Hunt Scenario by a Team of UAVs **42**

6 Cooperative Transport of Large Objects by a Pair of Unmanned Aerial Systems using Sampling-based Motion Planning **67**

7 Documentation of Dark Areas of Large Historical Buildings by a Formation of Unmanned Aerial Vehicles Using Model Predictive Control **76**

8 Autonomous Reflectance Transformation Imaging by a Team of Unmanned Aerial Vehicles **85**

9 Conclusion **94**
9.1 Future work 95

A Author's publications **96**
A.1 Thesis-related author's publications 96
A.1.1 Thesis core publications 96
A.1.2 Other thesis-related publications 97

B SCI Citations of author's work **99**

Bibliography **106**

Abbreviations

DARPA	Defense Advanced Research Projects Agency
FSM	Failure recovery and Synchronization jobs Manager
GNSS	Global Navigation Satellite System
GPS	Global Positioning System
IEEE	Institute of Electrical and Electronics Engineers
MAV	Micro-scale Aerial Vehicle
MBZIRC	Mohamed Bin Zayed International Robotics Challenge
MPC	Model Predictive Control
MRS	Multi-robot Systems Group
UAV	Unmanned Aerial Vehicle
UGV	Unmanned Ground Vehicle
RAS	Robotics and Automation Society
ROS	Robot Operating System
RTI	Reflectance Transformation Imaging technique

Chapter 1
Introduction

Research on Unmanned Aerial Vehicles (UAVs) is one of the most rapidly developing fields in mobile robotics. UAVs are becoming especially useful and widely used in numerous applications due to their potential for rapid deployment and their ability to reach locations that are difficult or dangerous for humans to access [25]. While fixed wing UAVs perform stable flights better at high speeds, long ranges, and extended flight times, rotary wing UAVs (such as the popular multi-rotor helicopters) benefit from high maneuverability, vertical take offs and landings, and better flight performance in cluttered environments or close to obstacles while being able to hover at a desired position in a 3D environment. With advances in technology, the acquisition costs for a single robot are decreasing. These lower acquisition costs have made it possible to deploy more robots to fulfill a given task. The deployment of a group of robots can decrease the time required for fulfilling a given task and increase the robustness of the system w.r.t. potential vehicle faults. In some cases, they may even complete missions that cannot be solved by a single vehicle, such as delivering large objects with weights exceeding the maximum payload of the individual robots.

In time-sensitive scenarios, such as disaster response, first responders face increased technical challenges, including rugged terrain, unstable structures, degraded environmental conditions, severe communication constraints, and extensive operation areas. To ensure the safety of human responders, autonomous robots such as UAVs are well suited for this task. Despite progress in the autonomy and reliability of UAVs, they are still most often teleoperated by a pilot while helping on site after natural disasters. Teleoperated UAVs are used for various situations, e.g., providing assistance for cities hit by an earthquake [26], [27], finding victims in urban areas [28], localizing flooded areas [29], finding survivors during floods [30], and quickly localizing forest fires [31]. However, to fully exploit the potential of UAVs, it is necessary to move from teleoperated robots to autonomous robots that are able to perceive their environment, reliably localize and navigate within it, and mutually influence their state by interacting with objects of interest. Furthermore, the use of multi-UAV teams entails additional challenges, such as inter-UAV communication and multi-robot conflict resolution. Therefore, in order to achieve efficient, safe, and reliable cooperation, multi-UAV architectures that are robust to the loss of faulty teammates and flexible enough to integrate and accommodate new software modules in an easy manner are needed [32]. All of these points have motivated the research presented in this thesis.

To push autonomous robot research further, several international robotics competitions have been organized in recent years. These competitions provide the opportunity for the best international teams of this field to deploy their approaches for particular problems outside of laboratory environments with limited time for preparation. The Defense Advanced Research Projects Agency (DARPA) Subterranean (SubT) Challenge [33] and the MBZIRC [34] are two of the most prestigious competitions, both of which focus on multi-robot scenarios.

The DARPA SubT Challenge aims to develop autonomous robot systems that would augment operations underground. Subterranean environments can extend for many kilometers and include highly constrained passages, multiple levels, and vertical shafts. Therefore, this

Figure 1.1: Photos from the DARPA Subterranean Challenge in a gold mine in the Rocky Mountains close to Denver, Colorado, USA [22a].

environment presents significant challenges for situational awareness. Figure 1.1 shows photos from the SubT Challenge.

The second competition mentioned is the MBZIRC competition. The MBZIRC is composed of several challenges motivated by the intent to push technological and application boundaries in robotics beyond the current state of the art. These technological challenges include fast autonomous navigation in semi-unstructured, complex, dynamic environments with minimal prior knowledge, robust perception and tracking of dynamic objects in 3D, sensing and avoiding obstacles, Global Navigation Satellite System (GNSS) denied navigation in indoor-outdoor environments, physical interactions, complex mobile manipulations, and air-surface collaboration.

This thesis is a compilation of six core publications [1c], [2c], [3c], [4c], [5c], [6c]. The work included in this thesis proposes robust high-level planning architectures and motion planning methods for multiple UAVs working simultaneously. The presented work is motivated mainly by following three robotic scenarios where the autonomous UAVs can be effectively used.

- **Autonomous Search, Grasping, and Delivery of Objects by a Team of UAVs**

  Delivery applications using a UAV are a hot topic nowadays and the community has already proposed several working solutions. In the USA, Google's parent company, Alphabet, has developed an autonomous UAV delivery service known as the Wing [35]. The Wing is a small, lightweight fixed wing aircraft with a navigation system that can deliver small packages, including food, medicine, and household items. Another USA corporation, Amazon, has proposed several platforms for the autonomous aerial delivery of small packages [36]. Examples of such platforms include a fixed wing aircraft, a multi-rotor aircraft, and a combination of fixed wing UAV and multi-robot UAV. The last platform takes advantage of both types of UAVs, making the vertical takeoff and landing done with a multi-rotor UAV and the long-range stable flights of fixed wing UAV possible. In September 2019, researchers from the National University of Ireland were able to use a UAV to deliver diabetes medication from Galway to a remote location in the Aran Islands [37]. This was the first successful delivery beyond a visual line of

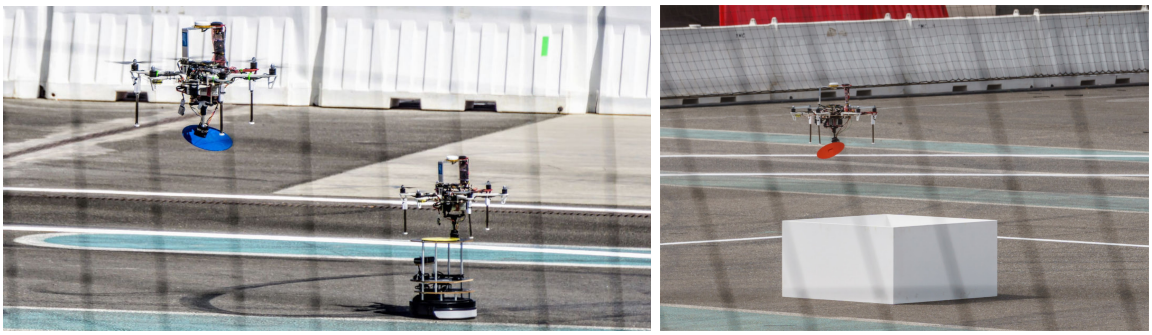


Figure 1.2: Photos from Challenge 3 of the MBZIRC 2017 in Abu Dhabi, UAE [3c].

sight diabetes UAV mission, demonstrating to the world that UAVs could carry medical supplies reliably. All of these works are focused on autonomous delivery, where the packages have to be placed on the UAV or removed from the UAV manually. However, the robotic community is interested in fully autonomous tasks composed of localization, acquisition, transport, and package drop-off. This interest in the community can be seen in Challenge 3 of the MBZIRC 2017, where a team of three UAVs had to find a set of static and moving colored ferromagnetic objects and deliver them to the target location. The deployment of a team of UAVs was motivated by the limited total mission time and the inclusion of large objects with weights exceeding the maximum payload of the individual robots (see Figure 1.2 for pictures from the MBZIRC 2017 competition). The system that won Challenge 3 of the MBZIRC 2017 is presented as part of this thesis[1].

- **Autonomous Aerial Surveys of Historical Building Interiors**

  A problematic task for historians and restorers working on heritage buildings, such as churches, is obtaining data to determine the state of objects of interest within the interior of historical structures. Typically, costly scaffolding must be built to acquire the data, but it is unfeasible to keep large scaffolding up for extended times due to regular services. This would, however, be necessary to study the long-term progress of restoration work. Some parts of these buildings could not even be reached by people for decades and would need to be inspected. In recent years, UAVs have been widely used in the heritage sector. This technology can be used to monitor outdoor archaeological sites, excavated objects, or for analysis of roof conditions, and likewise for monitoring artifacts, murals, and paintings within the interior of historical structures. UAVs can carry various sensors, such as high-resolution cameras or even laser scanners, allowing the capture of still images, videos, and data for detailed off-line analysis. Unfortunately, the interiors of these historical sites often have poor lighting conditions. Therefore, we propose a system designed for the inspection of difficult to access places under poor lighting conditions by a formation of autonomous UAVs (see Figure 1.3). This setup aims to fully autonomously apply the advanced illumination techniques that often used by historians and restorers for the manual inspection of the interiors of historical monuments. The use of autonomy brings several advantages that are barely manageable or even impossible to do with teleoperated UAVs. Such advantages include precise flight close to objects, access to places without an operator's direct line of sight, and the

[1] `http://mbzirc.com/winning-teams/2017/challenge3#viewwinner`

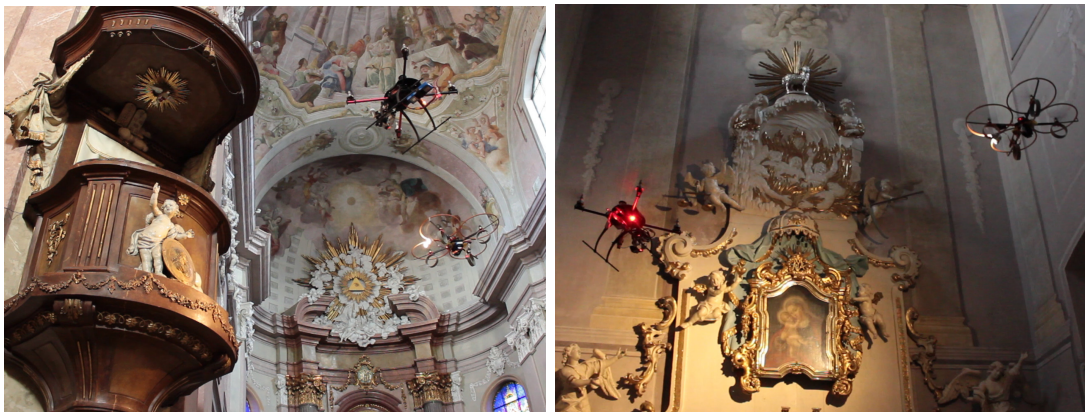


Figure 1.3: Photos from an inspection of the building interior of a historical church in Sternberk, CZE [6c].

possibility to perform the same flight repeatedly in order to help monitor the state of the object over time. In this thesis, we present the results of fundamental research on this topic. Currently, the obtained results are being used as the base of a system that has already been deployed for documentation in fourteen historical sites in the Czech Republic (CZE). The results from these deployments are publicly available at `http://mrs.felk.cvut.cz/dronument`.

- **Autonomous Firefighting**

  The last scenario is motivated by the use of robots for urban firefighting missions. This scenario has impelled the robotic community and industry, as can be seen by its selection as one of the challenges of the MBZIRC 2020 competition. A cooperative firefighting mission was the most complex task at MBZIRC 2020, requiring a team of robots to collaborate on a series of urban firefighting-related tasks in outdoor and indoor environments. For this task, three UAVs and one Unmanned Ground Vehicle (UGV) had to collaborate to autonomously extinguish a series of fires (real and simulated) in an urban building (see Figure 1.4). The fires were placed at various random locations at ground level in the arena (indoor and outdoor) and different heights of the building. The challenge can be divided into four separate sub-tasks: extinguishing interior fires by a UGV, extinguishing fires on the facade of the building by UAVs, extinguishing ground fires by UAVs, and extinguishing fires inside the building by UAVs. These tasks were meant to be solved in their full scope, including searching for fires with unknown positions, fire extinguishing, and cooperation among multiple UAVs and a UGV working in the same environment. The team deployment enables the use of a range of firefighting techniques (a fire-extinguishing agent or a fire blanket) and various platforms (UAV and UGV). In this thesis, we present a solution that was a part of the system which won the MBZIRC 2020 competition. Furthermore, the results of this research led to an industrial solution for the real firefighting using UAV that is now under development[2].

[2] `http://mrs.felk.cvut.cz/projects/dofec`

Figure 1.4: Photos from the firefighting challenge of the MBZIRC 2020 in Abu Dhabi, UAE. The left image shows fire extinguishing using a fire blanket. The right image displays UAV during the extinguishing of fire on the building facade by spraying a fire-extinguishing agent [14a], [18a].

1.1 Thesis outline

The rest of this thesis is organized in the following manner. In the next chapter, we will summarize the contributions of this thesis. In Chapters 3–8, a short introduction of individual core publications is followed by the manuscripts themselves. The thesis is concluded by Chapter 9, where the contributions of the thesis are discussed.

Chapter 2
Contributions

This chapter provides an overview to the author's contributions with the primary focus on four journal articles [1c]–[4c] and two conference proceedings in Web of Science [5c], [6c] that form the core of the thesis. This chapter contains three sections based on the relevance of the contributions of the individual core publications.

2.1 High-level planning architecture

When compared to a single robot, the deployment of multiple cooperating robots sharing the same environment presents several challenging aspects that must be solved. The most demanding aspect is communication. Communication between robots is required to fully access the capabilities of deployed multi-robot teams. This communication, for example, enables sharing observed maps between robots and their tasks. By sharing this information, the robots in the team are able to know the already-visited parts of the area, which could help speed up the exploration process during search & rescue missions.

Although communication brings many advantages, the technical requirements for working communication infrastructure are currently still a bottleneck. Therefore, the robotic community also provides approaches for teams of robots that do not rely on communication [38], or that only use explicit communication by observing other UAVs to achieve relative localization [39], [40]. A typical example of such an approach is the behavior-based swarm [41]. In these swarms, the collective behavior emerges from local information that can be directly obtained using onboard sensors. The locality allows achieving decentralized sensing and acting. Some other missions do not require direct coordination and cooperation between robots. In the case of communication drop out, it is beneficial for such missions to design a safety mechanism that can rely on topological properties of the tasks to achieve non-collision behavior.

For example, one of such missions is the urban firefighting challenge of the MBZIRC 2020 competition. As has been already mentioned, this challenge can be divided into four separate sub-tasks: extinguishing interior fires by a UGV [16a], extinguishing fires on the facade of the building by UAVs [14a], extinguishing ground fires by UAVs [18a], and extinguishing fires inside the building by UAVs [1c]. The last task of the challenge was primarily solved by the author of this thesis and is one of the core publications.

The employment of UAVs in firefighting has already been explored in several works. An obvious example of a situation where UAVs can prove beneficial is outdoor fire detection and monitoring. As reported in [42], a system of multiple UAVs can be used for automatic forest fire monitoring using visual and infrared cameras. Real experiments with forest fire monitoring in a national park have already been conducted by the Hungarian fire department [43]. The authors of [44] describe a task allocation strategy for the distributed cooperation of ground and aerial robot teams in fire detection and extinguishing. In [45], a UAV system is designed to extinguish a fire by dropping a fire-extinguishing capsule on it.

The use of UAVs for fighting fires has already been a topic in robotic competitions. [46] describes the design and implementation of a firefighting UAV for outdoor applications

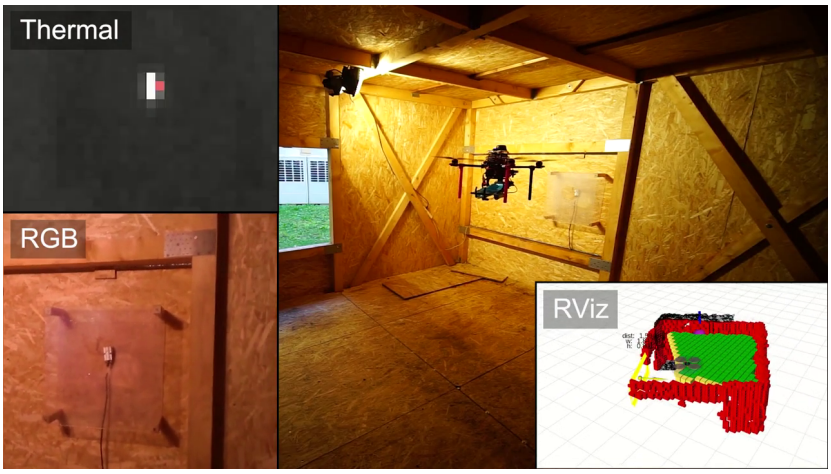


Figure 2.1: A photo of the fully autonomous UAV system developed for extinguishing fires inside buildings. The photo also shows the processed data from the onboard sensors of the UAV [1c].

designed specifically for the IMAV 2015 competition. The employment of UAVs could also prove to be beneficial and life-saving in urban environments. Studies have already been done on fire detection in urban areas using a thermal camera carried by a UAV [47]. UAVs capable of entering buildings through doors and windows will be especially helpful because of their ability to reach the target location much earlier than human firefighters. [48] contains the design of a semi-autonomous indoor firefighting UAV. The authors designed a fireproof, thermoelectrically-cooled UAV equipped with visual and thermal cameras, a collision avoidance module, and a first-person view system. However, to fully exploit the potential of UAVs in firefighting and to achieve reliable operation, the UAVs themselves need to be autonomous.

In the first core article of this thesis [1c], we present the methodology developed for the fully autonomous extinguishing of fires inside a building using a UAV. The overall system includes control and estimation of the UAV state, interior motion planning and exploration, window and fire detection and their position estimation, and fire extinguishing. One of the main contributions of this work lies in the precise control needed for flying through relatively small windows and for the precise spraying of a fire-extinguishing agent into a small opening representing the fire while using multiple sensory data to increase reliability. We have presented a novel multi-layer control pipeline that further enables precise localization and stabilization in an open space around a building, inside rooms with obstacles, and a smooth transition between these two environments (with GNSS and GNSS-denied). To achieve this new robotic challenge, a novel reliable high-level planning architecture was designed by the author of this thesis to be able to recover from failures and from a malfunction of the localization part. In addition, the author has contributed with a novel interior motion planning and exploration method that was the first method to solve the challenging task of collision-free GNSS-denied UAV motion planning using one sensory set and complete coverage of the interior by a different sensory set with a limited field of view. We refer to Chapter 3 for detailed information about the core publication [1c].

The following two core publications [2c] and [3c] utilize advantages of communication (if available) and provide a fallback solution in the case of communication dropout. This

challenge was successfully solved by the author of this thesis by designing a novel approach called the Failure recovery and Synchronization jobs Manager (FSM). The proposed FSM is crucial for managing all parts of the systems and coordinating all UAVs sharing the same workspace. The FSM was the first approach designed in order to achieve the reliability required for the deployment of UAVs in real-world conditions, which again requires the ability to recover from UAV failures and also from a malfunction of the localization and communication infrastructure. For example, for the first scenario of this thesis considered with the autonomous searching, grasping, and delivering of objects by a team of UAVs, the robots can easily collide with the objects being grasped due to gusting winds. In combination with the ground effect, this can create an unpredictable external force on the UAV in the final phase of the approach to an object. Such a collision could result in a UAV crash, deadlock, or an overturned object. Moreover, the malfunctions of UAV subsystems, such as camera dropouts, incorrect height measurements, gripper failure or gripper feedback failure, and imprecise object gripping, can be expected in demanding outdoor conditions. All these eventualities need to be considered by the FSM in order to enable undisturbed operation of the remaining robots in the event of a UAV failure, limited operation of a UAV with a faulty subsystem, or an unsuccessful or interrupted task. We refer to Chapter 4 and Chapter 5 for detailed information about these core publications [2c] and [3c], respectively.

The most relevant work that addresses multi-UAV applications of physical interaction with the environment considering the existence of or lack of communication is [38]. Their method is applied to autonomous structure assembly. The authors provide two solutions: an online planner when the communication is available and an offline planner when it is not. In the case of the offline planner, tasks must be assigned to the robots offline before the execution starts. Nevertheless, they do not force robots to follow the precomputed plan. They designed an online re-allocation strategy that can change the order of tasks and execute new ones not initially included in their plan to face unforeseen events during the construction process. However, the work [38] when compared to the [2c], [3c] is more focused on the task allocation problem and the results are presented using only simulations. Recently, other participating teams in the MBZIRC 2020 have published their approaches for multi-UAV coordination in autonomous construction, which was one of the challenges of the competition [32], [49]. Our MBZIRC 2020 team also participated in this challenge and won first place. Our approach to the autonomous construction problem is described in [13a].

2.2 Motion planning for multiple UAVs

The second scientific topic solved in this thesis is motion planning for multiple directly cooperating UAVs sharing the same workspace. The first core publication targeting this topic is [5c]. This work is focused on the cooperative transport of large objects by a pair of UAVs. The carrying of large objects using UAVs is a challenging task with several solutions having already been proposed [50]–[54]. The most relevant work [52] tackles the transportation of objects by rigidly attached quadrotors. However, the reliability and performance of this approach are difficult to compare to our proposed solution since the method is verified only in numerical simulations and the aspects of real-world deployment are not considered. In addition, it requires a known reference trajectory, which limits its usability in autonomous missions. The transport of large objects by multiple UAVs is addressed in [50], [51], where the

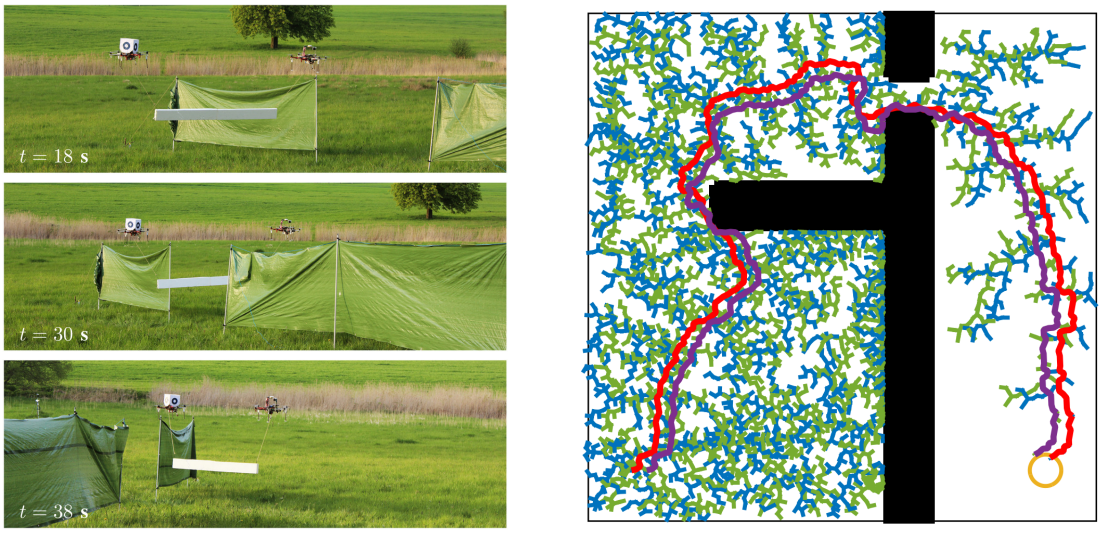
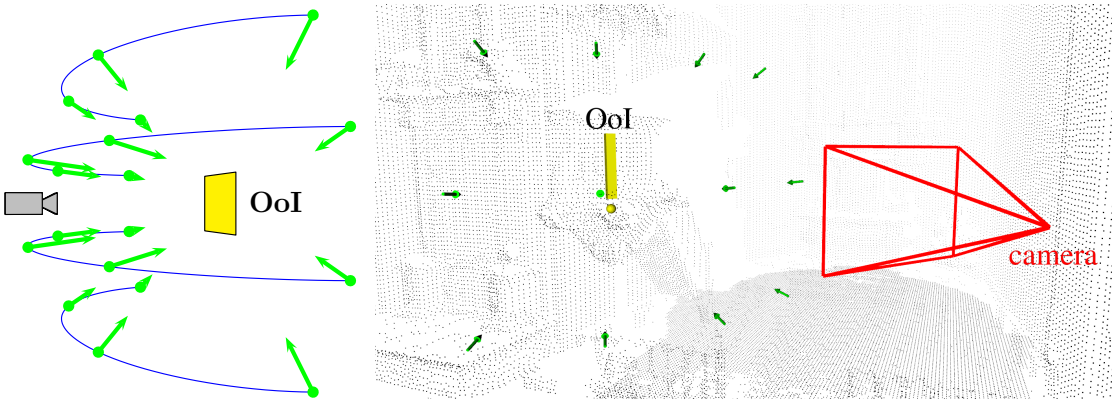


Figure 2.2: The left image shows snapshots from an outdoor experiment in which a large object is transported by a pair of UAVs through a narrow passage. The right image shows an example of a solution provided by a basic sampling-based motion planning algorithm [5c].

systems are designed for use in a simple indoor environment without obstacles and a motion capture system is used for team stabilization. A motion capture system was also used in [53], where the stabilization of a bar attached by ropes of different lengths to two UAVs is studied. Their system enables stabilization of the bar in a static position, but it does not allow transport in environments with obstacles. Transport by UAVs equipped with manipulators in an environment with obstacles is studied in [54]. In this approach, an estimate of the UAVs states are obtained by a motion capture system, and therefore possible real-world influences are not considered in the motion planning. Despite these recent works, the difficulty of this task in real-world conditions can be observed in results achieved by the leading robotic laboratories (selected from 143 applications) in the MBZIRC 2017, where none of the competitors was able to solve the given task cooperatively.

In [5c], we presented the first system designed for solving a cooperative transportation task by a UAV pair in environments with obstacles. The novelty of the proposed concept lies in planning a path that prefers to maintain the distance between the UAV pair as close as possible to the optimal for object carrying. However, it also enable temporary deviation when necessary. This prevents the pair carrying the object from experiencing undesired oscillations and increases the system's stability and robustness in real-world conditions where sensory data are affected by noise and UAVs may be subject to external disturbances. The core of the presented solution is a sampling-based motion planning algorithm, which was purposely designed to satisfy these needs. This novel method takes advantage of a guiding-based sampling of the configuration space [55]. At the same time, it includes cost-driven expansion to find an appropriate solution to the given task in terms of smoothness and quality of achieved paths, with respect to stability and reliability. Due to the cost function involved in the planning process, the UAVs are steered to operate in relative positions identified as the most appropriate for the given task. However, our proposed system enables temporary deviation from the desired shape of formation in the event that the surrounding environment requires



(a) illustration (b) experiment in real environment

Figure 2.3: The example of the generated set of Reflectance Transformation Imaging technique (RTI) goals marked with green dots and arrows. The yellow rectangle identifies the scanned object, and the blue curves indicate the horizontal splines, that represent possible positions of the RTI goals. Label OoI represents Object of Interest [4c].

it. This makes our solution applicable in a real-world environment with obstacles, where all other systems available at the time of publication would otherwise fail. The guiding-based process is essential to increasing the speed in which a solution to the motion planning task can be found. It also offers fast onboard re-planning in the event of detected environment alternation, which is also vital for the reliable real-world deployment of the system. We refer to Chapter 6 for detailed information about the core publication [5c].

The last two core publications selected in this thesis, [6c] and [4c], are motivated by the task of visual documentation and inspection of the interiors of historical buildings, with the research focus being on a formation of cooperating UAVs. The core publication [6c] presents a self-stabilized formation of UAVs employed for filming and visual inspection in dark conditions, where just one of the UAVs carries a camera and the neighboring UAVs each carry a source of light. The proposed novel control and planning approach enables setting the direction of the light sources dependently on the position of the camera, which makes it possible to realize two techniques with full autonomy — *Three point lighting* technique [56], [57] and *Strong side lighting* method [58]. Nowadays, these methods are often used by historians and restorers to manually inspect the interiors of historical monuments. It is a challenging problem to accomplish this with autonomous UAVs. We propose to define such methods as a multi-objective optimization problem in the Model Predictive Control (MPC) framework [59]. The MPC approach is used for control of the formation members while taking into account task objectives, as well as constraints of the formation flying (obstacle avoidance, mutual collision avoidance), low-level UAV stabilization (motion constraints), filming (limited camera field of view, keeping all UAVs out of the taken images), and illumination (providing the intensity of the lighting in a required range, keeping the recommended angle between the light and camera axes). Based on this theory, we have designed the first system enabling smart lighting by UAVs and experimentally verified it in the real conditions of historical objects. We refer to Chapter 7 for detailed information about the core publication [6c].

The last mentioned core publication [4c] presents a novel methodology to achieve a

Reflectance Transformation Imaging technique (RTI) [60], [61] autonomously by UAVs. RTI is a computational photographic method that captures the surface shape and color of a subject and enables interactive re-lighting from any direction in a software viewer, revealing details that are not visible to the naked eye. The input of RTI is a set of images captured by a static camera, each one under illumination from a different known direction. An innovative approach applying two UAVs to perform this scanning procedure was presented. [4c] presents three approaches to the generation of sequences of RTI positions and proposes an MPC-based framework for navigation of the UAVs between this generated sequence of positions (see Figure 2.3 with an example of the generated set of RTI goals). We refer to Chapter 8 for detailed information about the core publication [4c].

The formation stabilization and navigation technique in [6c] and [4c] arises from the theoretical work on leader-follower formation stabilization done by our lab at Czech Technical University [8a], [20a]. In leader-follower methods [62]–[64], one or more robots from the group are assigned the role of leaders while the rest are treated as followers. The task of the followers is to position themselves relative to the leader and maintain this relative position during a mission. The advantage of this method is its simplicity as the leader's behavior defines the behavior of the whole group. Furthermore, the leader-follower configuration is considered to be an energy-saving mechanism for aerial vehicles [65].

2.3 System for facilitation of transfer from fundamental achievements to real experiments

Finally, we mention one contribution of the author that is more technical than theoretical, but has positively influenced the scientific community in this field.

A well-designed simulation software makes it possible to test algorithms rapidly, design robots, perform regression testing, and train AI systems using realistic scenarios. Nowadays, the systems developed by the robotic community often rely on Robot Operating System (ROS) [66], which is an open-source set of software libraries and tools. Using ROS, complex tasks can be easily divided into smaller sub-tasks (nodes), which improves and clarifies the structure of the solution. One of the publicly available simulators is the Gazebo robotic simulator (Gazebo) [67]. The Gazebo simulator is compatible with ROS and allows to be used for software in the loop together with UAV autopilot firmware. This provides a very realistic testbed for UAVs and significantly simplifies testing the whole system. For this purpose, we have developed our simulation environment, which was made publicly available[1]. It makes use of the open-source Gazebo simulator and is set up for multiple different variants of our hardware UAV platforms (DJI f450, DJI f550, Tarot 650 sport, etc.). It can also easily be extended to a new hardware setup. All UAV hardware elements, including the Pixhawk flight controller[2], the actuators, and various sensors are simulated with high fidelity, so there is only a minimal difference between simulated flight and real-world flight. This ensures a smooth transition between simulation and reality, which significantly accelerates the deployment of new robotic methods and algorithms. Therefore, hardware experiments can be realized in a very short time and with fewer safety risks than with direct hardware verification.

[1] `https://github.com/ctu-mrs/simulation`

[2] The Pixhawk autopilot is an open-hardware and open-software architecture, which is advantageous for research in the field of aerial robotics [68].

Figure 2.4: Examples of different simulated environments (grass plane, forest, church interior, and cave) and UAV platforms.

Another significant advantage of our software is the possibility to easily deploy multiple UAVs simultaneously with different sensor settings in one environment. Figure 2.4 shows examples of different simulated environments and UAV platforms. This software has been thoroughly used and tested not only by our research group, but also by our research partners (University of Pennsylvania, New York University, CNRS Toulouse, University of Twente, University of Seville, and many more) during the last five years as a part of the Multi-robot Systems Group (MRS) system [10a]. For instance, the system helps research in projects focused on UAV mutual detection and localization [69]–[72], localization of sources of ionizing radiation using a group of small UAVs[3] [73]–[75], data collection [21a], [23a], [76]–[78], UAV swarms [40], [79], [80], or protection of critical infrastructures against rogue drones[4] [12a], [81]. This system is also being used for teaching and has already been used by more than 200 Master's degree or Ph.D. degree students from 50 different research groups attending the Institute of Electrical and Electronics Engineers (IEEE) Robotics and Automation Society (RAS) Summer School on Multi-Robot Systems in 2019[5] and 2020[6], or with study stays in our research group at CTU in Prague. This has allowed many robotic groups to conduct realistic verification of their scientific achievements and bridge the gap between theoretical and practical robotics.

The functionalities of methods and algorithms presented in these publications in which the author has contributed throughout the last five years were tested using the proposed software.

[3] http://mrs.felk.cvut.cz/radron
[4] https://eagle.one/en/
[5] http://mrs.felk.cvut.cz/summer-school-2019
[6] http://mrs.felk.cvut.cz/summer-school-2020

Chapter 3

Autonomous Firefighting Inside Buildings by an Unmanned Aerial Vehicle

In this chapter, we present the first core publication called *Autonomous Firefighting Inside Buildings by an Unmanned Aerial Vehicle* [1c] published in Q1 journal IEEE Access in 2021.

> [1c] **V. Spurny**, V. Pritzl, V. Walter, *et al.*, "Autonomous Firefighting Inside Buildings by an Unmanned Aerial Vehicle," *IEEE Access*, vol. 9, pp. 15 872–15 890, 2021

Here, we give a short overview of this 19-page paper. The paper itself is provided as the second part of this chapter.

The **introductory** section explains our motivation for developing an autonomous system for firefighting inside a building and highlights research contributions. This section lists related works by citing articles that focus on firefighting-related search & rescue missions and articles which deal with motion planning, control, and estimation of the UAV.

Section **autonomous system design** describes in detail parts of the proposed method. The main parts of the article discuss control and estimation of the UAV state in an outdoor and indoor environment, window detection and estimation, indoor motion planning and exploration, detection and estimation of the position of fires, fire extinguishing, and high-level behavior control. The high-level behavior control is the main content of this thesis, as it is the crucial component responsible for the robustness of the entire system by providing failure detection and recovery abilities.

The following **experimental results** section presents the results obtained by multiple deployments of the presented methodology in simulated and real-world environments. The proposed methodology was also selected as the core of an industrial firefighting UAV system using fire-extinguishing capsules, which supports the practical usability of the proposed fundamental research results.

IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received December 14, 2020, accepted January 7, 2021, date of publication January 20, 2021, date of current version January 28, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3052967

Autonomous Firefighting Inside Buildings by an Unmanned Aerial Vehicle

VOJTECH SPURNY, VACLAV PRITZL, VIKTOR WALTER, MATEJ PETRLIK, TOMAS BACA, (Graduate Student Member, IEEE), PETR STEPAN, DAVID ZAITLIK, AND MARTIN SASKA, (Member, IEEE)

Department of Cybernetics, Faculty of Electrical Engineering, Czech Technical University in Prague, 160 00 Prague, Czech Republic

Corresponding author: Vojtech Spurny (vojtech.spurny@fel.cvut.cz)

This work was supported in part by the Czech Science Foundation (GAČR) under Project 20-29531S, in part by the Research Center for Informatics Project under Grant CZ.02.1.01/0.0/0.0/16_019/0000765, in part by CTU under Grant SGS20/174/OHK3/3T/13, in part by the EU H2020 Project AERIAL CORE under Grant 871479, and in part by the Khalifa University via sponsorship in support of 15 selected teams for the MBZIRC 2020 competition.

ABSTRACT This paper presents a novel approach to autonomous extinguishing of indoor fires inside a building by a Micro-scale Unmanned Aerial Vehicle (MAV). In particular, controlling and estimating the MAV state, detection of a building entrance, multi-modal MAV localization during the outdoor-indoor transition, interior motion planning and exploration, fire detection and position estimation, and fire extinguishing are discussed. The performance of these elements, as well as of the entire integrated system, are evaluated in simulations and field tests in various demanding real-world conditions. The system presented here is part of a complex multi-MAV solution that won the Mohamed Bin Zayed International Robotics Challenge 2020 (MBZIRC 2020) competition, and is being used as the core of a fire-fighting Unmanned Aerial System (UAS) industrial platform under development. A video attachment to this paper is available at the website http://mrs.felk.cvut.cz/2020firechallenge-insidefires.

INDEX TERMS Unmanned aerial vehicle, autonomous systems, firefighting, mobile robots, rescue robots.

I. INTRODUCTION

Micro-scale Unmanned Aerial Vehicles (MAVs) are nowadays used in numerous applications due to their potential for rapid deployment and their ability to reach locations that are difficult or dangerous for humans to access [1]. Despite advances in the autonomy and the reliability of MAVs, they are most often still teleoperated by a pilot while helping on site after natural disasters. Teleoperated MAVs are used for various situations, e.g. for providing assistance for cities hit by an earthquake [2], [3], finding victims in urban areas [4], localizing flooded areas [5], finding survivors during floods [6], and quickly localizing forest fires [7]. Further examples of robots assisting in search and rescue missions are presented in [8].

The MAVs deployed in the applications mentioned above operate at high altitudes, where no obstacles can be encountered and Global Navigation Satellite System (GNSS) localization is reliable. However, to fully exploit the potential of MAVs assisting in disaster response tasks, it is necessary to move from teleoperated robots to autonomous robots that perceive their environment, can reliably localize and navigate in it, and furthermore, can influence their state by interacting with objects of interest. Autonomous MAVs have already been tested for use in locations where teleoperated MAVs cannot operate, e.g. coal mine tunnels [9], which can be dangerous to access after natural disasters such as earthquakes or

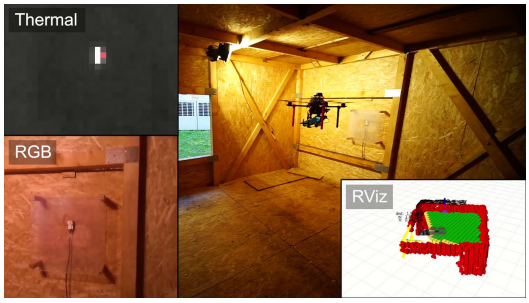


FIGURE 1. The proposed system uses processed data from the onboard sensors of the MAV towards the goal to extinguish fire inside the building.

The associate editor coordinating the review of this manuscript and approving it for publication was Juan Liu.

gas fires. A multi-modal mapping unit [10] can be attached to an MAV to provide autonomous exploration of GNSS-denied dark environments. A small agile MAV can be deployed as a first responder [11] of a rescue team to assess the situation in mines, to evaluate the risk of human rescuer injury and, most importantly, to find visual cues about the location of possible survivors in order to direct further rescue operations. During search and rescue operations, human rescuers underground often risk exposure to noxious gases. To reduce such hazards, a system described by [12] deploys an MAV with sensors capable of detection and measurement of the concentration of such gases. Multiple gas detecting MAVs deployed in parallel can be used for precise localization of the gas source [13]. Operations in places that are in the proximity of a source of radiation, such as the interior of a nuclear power plant, have to be planned with limited exposure time of human workers. The assistance of autonomous MAVs is therefore valuable in localizing the source of radiation [14], or in finding survivors [15], without risking prolonged exposure of human rescuers.

The approaches mentioned above were designed only for indoor environments or only for outdoor environments. However, the challenge and the novelty of the approach presented here is in the required transition from the open space around the building into the confined space of the rooms. These environments require different localization and state estimation techniques.

A cooperative firefighting mission called the Fire Challenge was the most complex task at MBZIRC 2020.[1] This challenge was motivated by the use of robots for urban firefighting, and it required a team of robots to collaborate on a series of urban firefighting-related tasks in outdoor and indoor environments. In this challenge, three MAVs and one Unmanned Ground Vehicle (UGV) had to collaborate to autonomously extinguish a series of fires (real and simulated) in an urban building. The fires were placed at various random locations at ground level in the arena (indoor and outdoor), and at different heights of the building. The challenge can be divided into four separate sub-tasks: extinguishing interior fires by a UGV, extinguishing fires on the facade of the building by MAVs, extinguishing ground fires by MAVs, and extinguishing fires inside the building by MAVs. These tasks were meant to be solved in their full scope, including searching for fires with unknown positions, fire extinguishing, and cooperation among multiple MAVs and a UGV working in the same environment. The deployment of such a team, as opposed to a single unit, was motivated by the requirement for minimal total mission time, as time is a critical factor for eliminating fire spreading and for saving people in real scenarios. Team deployment also enabled the use of a range of firefighting techniques (a fire-extinguishing agent or a fire blanket) and various platforms (MAV and UGV).

This paper addresses what we consider to be the most challenging task of the third challenge of the MBZIRC 2020 competition – fire extinguishing inside a building (see Fig. 1). The work presented here contributed not only to the MBZIRC 2020 challenge and to the firefighting mission but it also benefits other MAV applications. Research on indoor-outdoor transition through the narrow space of small windows and precise multi-sensor based servoing is important for a wide range of MAV challenges that are being tackled nowadays.

[1] http://www.mbzirc.com/

A. PROBLEM DEFINITION AND REQUIREMENTS ON THE PLATFORM

To solve the task of fire extinguishing inside a building, we assume that the size of the MAV platform, including the propellers, is limited by the width of the windows through which the platform has to fly. Of course, this width will be unknown during the deployment of the system in a real firefighting scenario. However, the organizers of the MBZIRC 2020 competition specified the window size to dimension of 2 m. Choosing the right platform size is crucial to task performance, as a smaller platform allows for a larger margin of error of the localization and control systems. On the other hand, a smaller MAV can carry less fire-extinguishing agent, and cooperation among multiple agents may be needed to extinguish a single fire.

We expect that the MAV will be equipped with a flight controller that commands the Electronic Speed Controllers (ESCs) to drive the brushless motors propelling the MAV, based on angular rate commands from an onboard computer. Furthermore, this flight controller should contain a set of sensors, such as accelerometers, gyroscopes, barometers, and magnetometers, and should provide them to the onboard computer for MAV state estimation. This onboard computer should provide sufficient computational power to solve all the required onboard processing tasks, in addition to besides general MAV control, state estimation, and collision-free motion planning.

For outdoor flying capability, the MAV has to be equipped with a GNSS receiver. However, the precision of the position data derived from satellite-based positioning systems can drift in the proximity of tall structures such as buildings, and this may block the visibility of some of the satellites, or may reflect the signal. The most fitting sensors that can be deployed to avoid a possible collision with the building are 3D LIDARs, thanks to their high information density and precise measurements of the obstacle distance. However, these devices are still relatively expensive and heavy. The proposed system therefore requires the MAV to be equipped with two complementary sensors — 2D LIDAR and a stereo camera. These sensors were selected because the data they provide can also be used for MAV control, for state estimation, and for collision-avoidance inside the building. Furthermore, this data is useful for correct detection of the window, for planning a collision-free trajectory through it, and also for estimating the position of detected fires.

The proposed system requires a thermal camera, or rather, a compound sensor consisting of multiple thermal cameras

for detecting fires. These cameras need to be arranged in such a way that the MAV flying at a safe distance from the wall will cover the whole wall, from the floor to the ceiling, with their Field Of Views (FOVs) to minimize the chance of missing a fire source. The MAV should be further equipped with two laser rangefinders. One faces downwards to measure the distance to the ground, and the second faces upwards to measure the distance to the ceiling when the MAV is inside the building. Data from these sensors is used for MAV altitude estimation and to help ensure safe flight. To extinguish the fires, the MAV has to be equipped with a water bag and with a pump that can force water through a nozzle mounted on the front of the MAV. To reduce the weight, the nozzle can be rigidly attached to the MAV, since it does not have to be actively stabilized.

We assume that the GNSS signal is available only outside the building, and not inside. The building interior contains unknown obstacles (*e.g.* a bed, a TV with a table, or a dining table), so that the system for fire extinguishing inside the building meets realistic assumptions. Lastly, we considered that a direct line of sight to the MAV would not be maintained during the whole mission, especially after entering the building, and that teleoperation through the base station would not be possible. Therefore, the task has to be solved completely autonomously, using only the onboard equipment of the MAV.

B. RELATED WORK AND CONTRIBUTION

Employing MAVs in firefighting has already been explored in several works. An obvious example of a situation where MAVs can prove beneficial is outdoor fire detection and monitoring. As reported in [16], a system of multiple MAVs can be used for automatic forest fire monitoring using visual and infrared cameras. Real experiments with forest fire monitoring in a national park have already been conducted by the Hungarian fire department [17]. The authors of [18] describe a task allocation strategy for distributed cooperation of ground and aerial robot teams in fire detection and extinguishing. In [19], an MAV system is designed to extinguish a fire by dropping a fire-extinguishing capsule on it.

MAVs fighting fires have also already been a topic for robotic competitions. [20] describes the design and implementation of a firefighting MAV for outdoor applications designed specifically for the IMAV 2015 competition. The employment of MAVs could also prove to be beneficial and life-saving in urban environments. Studies have already been done on fire detection in urban areas using a thermal camera carried by an MAV [21]. MAVs capable of entering buildings through doors and windows will be especially helpful, because of their ability to reach the target location much earlier than human firefighters. [22] contains the design of a semi-autonomous indoor firefighting MAV. The authors designed a fireproof, thermoelectrically-cooled MAV equipped with visual and thermal cameras, a collision avoidance module, and a first-person view system. However, to fully exploit the potential of MAVs in firefighting and to achieve reliable operation, the MAVs themselves need to be autonomous.

MAVs autonomously entering a building through a window has already been partially explored in the literature, using various approaches with varying levels of experimental verification. In [23], the authors used RGB camera images and 2D LIDAR data for window detection and tracking, visual servoing while approaching the window, and potential field-based planning for the fly-through itself. However, their approach for window detection requires an operator to manually select a point of interest in the RGB image, and was verified only by simulations. This approach was further extended and verified in real-world experiments with flight through a wooden frame in [24]. However, the experimental verification only consisted of manually guiding the MAV in front of the window, autonomous flight through the window, and immediate manual landing. [25] utilizes stereo image pairs for detecting and estimating a window that can potentially be used by an MAV for entering a building. However, the proposed algorithm was verified only on data captured using a hand-held stereo rig. [26] deals with window detection from an RGB-D camera along with the generation of an optimal trajectory to a point in front of the window, but the approach was only verified in simulations. [27] focuses on cooperative control of an ornithopter MAV using visual servoing for narrow passage traversal. They demonstrated their approach in a real experiment with a small MAV flying through a narrow wooden frame. This approach requires a ground station continuously observing the scene throughout the flight. [28] deals with state estimation, control, and planning for an aggressive flight by an MAV through a narrow window tilted at various angles. State estimation is done based on visual camera images and an Inertial Measurement Unit (IMU). However, the position of the window needs to be known beforehand. Similarly, [29] focuses on an aggressive flight of an MAV through narrow gaps tilted at various angles employing a forward-facing fisheye camera for gap detection. A black-and-white rectangular pattern was used to simplify the detection of the gap. The approach for window detection and autonomous entering of buildings proposed in this paper was designed specifically for reliable performance under real-world conditions, and to function as a part of a complex autonomous system without input from a human operator. As such, the proposed approach was extensively verified in complex real-world experiments and therefore surpasses previous approaches both in the degree of autonomy under real-world conditions and in the complexity of its experimental verification.

Transition between an indoor environment and an outdoor environment creates the necessity to combine different localization methods in a single flight. [30] describes a system combining visual and laser odometry with IMU, using an Extended Kalman Filter (EKF) for flight in both indoor and outdoor environments. The authors use one common filter for fusing measurements from laser scan matching and from correlation-based visual odometry. The data source that is currently fused is determined by its variance. Fusion in one

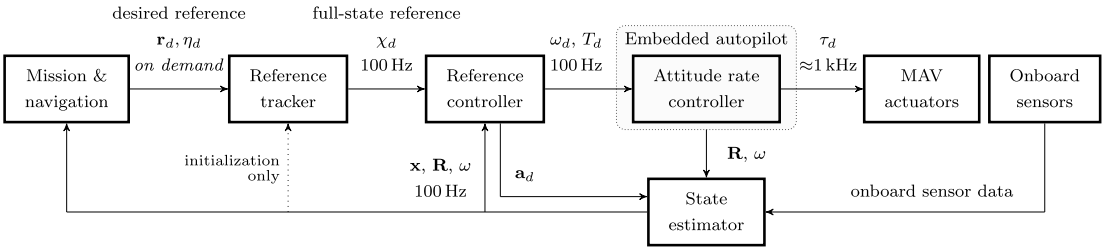


FIGURE 2. A diagram of the control system architecture. ***Mission & navigation*** software supplies a position and heading reference ($\mathbf{r_d}$, η_d) to the reference tracker. The ***Reference tracker*** creates a smooth and feasible reference χ_d for the reference feedback controller. The feedback ***Reference controller*** produces the desired thrust and angular velocities (T_d, ω_d) for the embedded ***Attitude rate controller***. The ***State estimator*** fuses control input $\mathbf{a_d}$ with data from the onboard sensors to create an estimate of the MAV translation and rotation ($\mathbf{x}$, $\mathbf{R}$, ω). For a more detailed description of the whole control pipeline, see [32].

common filter is problematic when the reference frames of the two odometry sources are not coincident, *e.g.* due to imprecise extrinsic calibration. Moreover, in the case of GNSS and LIDAR odometry, the frames of reference are inherently different. We propose fusing each type of measurement in its own separate filter and then choosing the better output to close the control feedback loop. [31] describes an approach that uses depth image processing for visual odometry capable of navigating MAVs during indoor and outdoor flight, and during transfers between these two types of flight. The solution relies on stereo camera depth estimation, which is much less precise than direct distance measurements using a laser sensor. Our solution offers a higher level of autonomy, as the whole mission is governed by a mission control state machine while in [31] the MAV is controlled by waypoints manually entered by an operator.

To sum up, the contributions of this paper are in the complexity and the reliability of the proposed system, which includes indoor outdoor transition, interaction with the environment based on vision from thermal cameras, precise MAV stabilization and control for safe flight through window, and for firefighting. Furthermore, the paper proposes a new approach for handling data from multiple sensors to robustly obtain a single state estimate – MAV pose estimation, height estimation, relative window pose estimation, and relative pose of the fire target estimation. All of these estimates are crucial for safe autonomous flight in complex MAV missions, and the proposed redundancy by using various sensors is necessary for achieving reliability required for industrial applications.

II. AUTONOMOUS SYSTEM DESIGN

The proposed system components are described in this section. Note that the entire system is run on the onboard PC only, allowing for full autonomy without any control station or teleoperation needed.

A. CONTROL AND ESTIMATION OF THE MAV STATE IN OUTDOOR AND INDOOR ENVIRONMENTS

One of the main contributions of this paper is a system that allows precise control for flying through relatively small windows and for inserting water into a small opening of a measurement device using multi-sensor control feedback. Additionally, in a real scenario, precise placement of the fire fire-extinguishing agent is crucial for mission success. Another important aspect of the system is the MAV state estimation approach, which allows precise localization and stabilization in the open space around the building, inside rooms with obstacles, and a smooth transition between these work-spaces.

The MAV is controlled by the novel multi-layer control pipeline, depicted in Figure 2, which was suited for the proposed system using the general control framework presented in [32]. The desired trajectory reference is supplied by higher-level motion planning modules that are specialized for each mission phase, as described in II-F. The reference is first processed by the *Reference tracker* [33], based on model predictive control to obtain a smooth and feasible reference for the *Reference controller*. The tracker also imposes constraints on the MAV states to prevent fast and aggressive motions, which are undesirable when navigating constrained indoor environments. The *Reference controller* uses the processed reference to provide SE(3) geometric state feedback control [34] of the translational dynamics and the orientation of the MAV. This type of controller achieves minimal control errors, which allows precise window flythrough. The attitude rate and thrust commands generated by the *Reference controller* are sent to the embedded *Attitude rate controller* in the flight control unit of the MAV, which controls the speed of each motor, using ESCs. The feedback loop of the *Reference controller* is closed by the *State estimator*, which fuses data from onboard sensors with the MAV altitude to obtain a precise and reliable state estimate for both the indoor and the outdoor phase.

The state estimation process uses Kalman filtering to estimate the 3D position of the MAV and its heading angle, along with their respective first and second derivatives. The MAV state is divided into lateral, altitude, and heading parts. Such decoupling facilitates tuning of the filters. Smaller system matrices save computation resources, allowing for running multiple filters in parallel. All active filters are grouped in a filter bank, visualized in Figure 3, from which the best filter

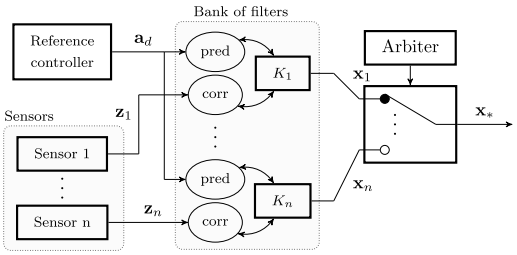
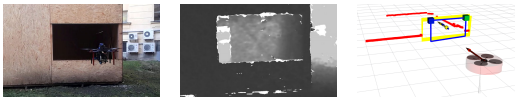


FIGURE 3. The bank of filters $K = \{K_1, \ldots, K_n\}$. The prediction step is driven by the desired acceleration $\mathbf{a}_d$. The correction step is triggered asynchronously as sensor measurements $\mathbf{z}_i$, where $i = \{1, \ldots, n\}$, arrives. The output hypothesis $\mathbf{x}_*$ is chosen by the arbiter.



(a) View from an external camera. (b) Depth image from the onboard camera. (c) Visualization of the data.

FIGURE 4. Example of window detection using the stereo camera and 2D LIDAR. The red squares in the visualization represent the LIDAR data, the blue rectangle represents the depth detection, and the yellow rectangle represents the filtered estimate.

for the current situation is used to close the feedback loop of *Reference controller*.

Since the fire-extinguishing mission consists of two phases (outdoor and indoor), the bank of filters for the lateral axes contains one for each phase. Both filters have the same three-state model with the desired acceleration from *Reference controller* on the input. The difference between the filters is in the sensor measurement used to correct the state in the update step of the Kalman filter iteration. The outdoor filter uses position corrections from GNSS and heading corrections from magnetometers. Inside the building, GNSS cannot be used. Both position corrections and heading corrections are therefore provided by the Hector SLAM [35] algorithm, using 2D LIDAR. In general, the bank of filters may fuse multiple sensors and localization techniques for state estimation (e.g. 3D LIDAR, VIO, Optic Flow). One of the simultaneously running filters is always selected as the main estimator, depending on the reliability of the filter state estimation (estimation covariance) and based on consideration of the current environment. For the task solved in this paper, the high-level planner changes the main lateral estimator during transitions between indoor and outdoor phases, just before flying through a window. This approach prevents measurements from the sensor that are inappropriate for the current surroundings to corrupt the state estimate used for feedback control. Greater reliability is thus ensured during the window flythrough, thanks to lower position drift. The estimation framework is also responsible for synchronously broadcasting the change of main estimator, so that the tracker and the controller can react by updating their internal state accordingly. The switch of the main estimator is smooth and seamless without producing any spikes in the controller output, which makes it virtually unnoticeable.

The altitude estimation fuses data from a barometer with measurements from a laser rangefinder. The rangefinder measures the height above the terrain, which can result in sharp changes in height when flying above objects protruding from the ground plane. Measurements of this type are declined by a median filter when the tall object is visible for less than 1 s and, in combination with the barometer data fusion, a smooth altitude estimate suitable for control is obtained. In the case of a height measurement jump lasting for more than 1 second, the measurement is offset by the difference from the original height. Up until this time, the median filter suppresses reaction to the measurement change.

B. WINDOW DETECTION AND ESTIMATION

In order to enter the target building, a suitable entrance needs to be detected and its position needs to be continuously estimated during the entire flythrough. For this purpose, depth data from the stereo camera and 2D LIDAR data are utilized. The depth data can provide complete information about the 3D position, the orientation, and the size of the window. However, these detections contain significant noise and the FOV of the stereo camera is limited. On the other hand, the 2D LIDAR data is very precise and 2D LIDAR is capable of seeing the window during the whole flythrough maneuver, regardless of the MAV orientation. The main drawback of 2D LIDAR is that it only provides information about the window in the 2D horizontal plane. The proposed approach offers multiple modes of estimation. In *depth + lidar* mode, the depth data is fused with the 2D LIDAR data with no apriori-known information required. In *lidar + apriori* mode, only 2D LIDAR detections and apriori-known information about window size and altitude are used for window detection. This mode was used to simplify window detection for the competition, in which information about window size and altitude was made available beforehand. Both modes work only in the case that the window is not obstructed.

Window detection from the depth data is described in Algorithm 1. The algorithm takes the 1280×720 depth image published by the stereo camera and downsamples it by a factor of 8 to reduce the computational demands on the CPU. The algorithm then searches for contours in the image and tries to fit quadrilateral shapes with certain parameters to the data. After identifying such quadrilaterals, a check is performed to ensure that the detected shape is an opening and not a protrusion. The algorithm starts from the center of the quadrilateral, where it generates a Region of Interest (ROI) such that its aspect ratio is the same as the initial quadrilateral and the shorter size is two pixels. It then checks if the depth of all the pixels within this ROI is greater than the plane of the initial quadrilateral. Then it iteratively expands this ROI and repeats the check until the check fails. Afterwards, the corners of the last ROI are projected to the plane of

Algorithm 1 Detection of Window From Depth Data

Input: Raw depth image $\mathcal{I}_{raw}$
Output: List $\mathcal{D}_{depth}$ of detected windows

function Detect_windows_depth($\mathcal{I}_{raw}$)
  $\mathcal{I}_{down} \leftarrow$ Downsample($\mathcal{I}_{raw}$)
  $\mathcal{C} \leftarrow$ Find_contours($\mathcal{I}_{down}$)
  for $c \in \mathcal{C}$ **do**
    $q \leftarrow$ Fit_quadrilateral(c)
    if $q \neq \varnothing$ and Is_hole(q) **then**
      Add q to $\mathcal{D}_{depth}$
  return $\mathcal{D}_{depth}$ ▷ Detections are passed to the LKF

the initial quadrilateral. Then the size of the quadrilateral formed by the projected points is compared with the size of the initial quadrilateral. If it is above a certain ratio, the new quadrilateral is accepted as a traversable window. Otherwise, the detection is discarded.

Window detection from 2D LIDAR data is described in Algorithm 2. Firstly, a combination of line extraction algorithms is utilized to identify possible window edges. The Successive Edge Following algorithm [36] is used for detecting window edge candidates $\mathcal{E}_1$ based on sudden changes in 2D LIDAR measurements. The algorithm parses the original scan into a set of segments $\mathcal{S}_1$ by splitting the scan in places where the difference between two consecutive measurements exceeds a predefined threshold. The endpoints of the segments $\mathcal{S}_1$ are used as window edge candidates $\mathcal{E}_1$. The set of segments $\mathcal{S}_1$ is then passed to the Iterative End-Point Fit algorithm [36]. The algorithm fits a line through the endpoints of each segment and splits the segment into two sub-segments at the point most distant from the line, if the distance exceeds a predefined threshold. This process enables us to detect the window edges in the form of a corner protruding towards the MAV, which occurs when a wall or an obstacle is located right next to the window inside the building. Depending on the mode of estimation, the final detections are then produced either by linking the window edge candidates $\mathcal{E}$ to existing window estimates $\mathcal{W}$ initialized based on depth detections, or by standalone detection based on apriori-known information $\mathcal{A}$ describing window size and altitude.

The Linear Kalman Filter [37] is used for fusion of the individual detections and for filtering out measurement noise. The state $\mathbf{x}$ of the Kalman filter describing a single window is defined as

$$\mathbf{x} = \left[c_x, c_y, c_z, \phi, w, h\right]^T, \tag{1}$$

where c_x, c_y, c_z are Cartesian coordinates of the window center, $\phi \in [-\pi, \pi]$ is the angle between the projection of the normal vector of the window to the xy-plane and the x-axis (i.e., rotation around the z-axis), w is the width, and h is the height of the window. It is assumed that the window is not tilted and is perpendicular to the ground plane. The position and the orientation of the window are specified in

Algorithm 2 Detection of Window From 2D LIDAR

Input: List $\mathcal{P} = \langle \boldsymbol{p}_1, \ldots \boldsymbol{p}_n \rangle$, where $\boldsymbol{p}_i$ are points obtained from a single laser scan; $mode \in \{depth + lidar, lidar + apriori\}$ - selected mode of estimation; (optional) list $\mathcal{W}$ of existing window estimates; (optional) list $\mathcal{A}$ of apriori information
Output: List $\mathcal{D}_{lidar}$ of window edge pairs

function Detect_windows($\mathcal{P}$)
  $\mathcal{E}_1, \mathcal{S}_1 \leftarrow$ SEF($\mathcal{P}$) ▷ Successive Edge Following
  $\mathcal{E}_2, \mathcal{S}_2 \leftarrow$ IEPF($\mathcal{S}_1, \mathcal{P}$) ▷ Iterative End-Point Fit
  $\mathcal{E} \leftarrow \mathcal{E}_1, \mathcal{E}_2$
  if $mode = depth + lidar$ **then**
    $\mathcal{D}_{lidar} \leftarrow$ Link_edges_to_estimates($\mathcal{E}, \mathcal{W}$)
  else if $mode = lidar + apriori$ **then**
    $\mathcal{D}_{lidar} \leftarrow$ Standalone_detection($\mathcal{E}, \mathcal{A}$)
  return $\mathcal{D}_{lidar}$ ▷ Detections are passed to the LKF

a global coordinate frame and therefore all the state vector elements are modeled as static. Figure 4 shows an example of window detection for a mock-up building constructed at the Czech Technical University in Prague. The figure contains an external view of the MAV hovering in front of the window, a depth image from the onboard camera, and a visualization of the data.

C. INDOOR MOTION PLANNING AND EXPLORATION

The position of the fire is unknown (in the competition and usually also in a real application) before the mission and therefore the interior must be explored to find its location. Obstacles can be detected using data from the stereo camera and from 2D LIDAR. We store the information about the environment around the MAV in an occupancy grid within a three-dimensional buffer that slides along with the MAV. Our buffer is a modification of the implementation of a buffer that has been developed as a part of the *Ewok* system [38]. Originally, the buffer was in the shape of a cube, with the same number of blocks in all axes. However, the height of a room is usually much smaller than its horizontal dimensions. Therefore, we modified the buffer to be able to specify its z-axis dimension separately. For the experiments, the buffer size was set to $128 \times 128 \times 32$ voxels with resolution of 0.1 m. This improves the times for inserting the data from the sensors and for recomputing the buffer in the case of MAV movement. See an example of a visualization of the occupancy grid in Figure 5(a).

The trajectory planning in our case is done using an A* planner that finds a local plan in a three-dimensional grid with a predefined minimal allowed distance from the obstacles (the grid has the same resolution as the occupancy grid). This plan is further processed by removing redundant points, in order to smooth out the overall path. The points that can be removed are those where the path between the previous point and the next point in the plan is still safe. This is done iteratively until no points can be removed from the plan. Because the

(a) A visualization of an occupancy grid stored in a three-dimensional buffer. The size of the grid is 128×128×32 with resolution of 0.1 m.

(b) A visualization of a two-dimensional grid with information about this already scanned space. The size of the grid is 128×128 with resolution of 0.15 m.

FIGURE 5. Examples of occupancy grids around the MAV used for motion planning and exploration inside the building. The red elements denote obstacles, the green elements show the already scanned space, and the yellow elements denote frontiers for exploration.

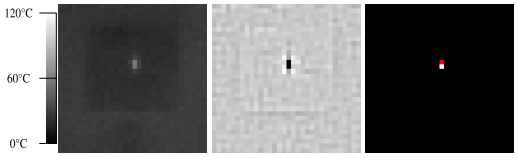


FIGURE 6. Thermal vision outputs. From left to right: Thermal camera view scaled from 0 to 120 °C, differential image produced with the Laplacian operator (scaled between the two extreme values of the image), and thresholded detection with the rounded centroid in red.

stereo camera at the front has a limited FOV, movement in any direction is allowed only when the MAV is facing in that direction. The plan is then sampled according to the permissible dynamics of the MAV for planning procedure and send to the reference tracker. The dynamic of the MAV for the planner was set for all deployments presented in Section III to a speed of $0.3\,\mathrm{m\,s^{-1}}$ and a heading rate of $0.3\,\mathrm{rad\,s^{-1}}$. This is within the capabilities of the MAV platform that had flying dynamic constraints set to a speed of $2\,\mathrm{m\,s^{-1}}$ and a acceleration of $10\,\mathrm{m\,s^{-2}}$. This ensures that the processed trajectory produced by the reference tracker will not differ from the plan by an unsafe amount.

One of the primary goals for successfully completing the mission is to find the fires. Using the planning presented here, we can fly without collision inside the building. However, it is necessary to specify the position that we want to reach. To ensure that each room is completely scanned for fire sources, a novel exploration algorithm had to be designed. The whole exploration process is described in Algorithm 3. The proposed approach differs from state-of-the-art exploration methods, which are designed to build a map in which the robot is localized. In our case, one set of sensors is applied for simultaneous localization and mapping, and a different set is applied for fire source localization. Moreover, the FOV of the two sensory sets differs significantly, requiring different exploration strategies. The exploration strategy relies on information about which parts of the interior have already been seen by the thermal cameras. The part that has already been visited (i.e. visually scanned) is the space that was within the FOV of the thermal cameras and is closer than the maximum detection distance. We assume that the height of the room is completely covered by the vertical FOV of the thermal cameras. The exploration can therefore be simplified and solved as a two-dimensional problem, where the z-coordinate of the goal position is set to a constant flying height (the z-coordinate of the center of the opened window). For this purpose, a two-dimensional occupancy grid is built from the incoming sensor data. The horizontal FOV of the thermal cameras is projected to this grid, where all elements of the grid located within this field and unobscured by obstacles are updated. When an element has been updated thirty times (the replanning was set to 10 Hz, meaning that the block has been observed for 3 s), the block is marked as scanned. Frontiers - elements that are marked as scanned and that have at least one neighboring unscanned element - are then candidates for goal positions for the planning. From these frontier elements, we select the element that can be scanned the earliest and which has at least twenty unscanned elements in its proximity. If the selected goal position can be seen from the current position of the MAV merely by changing the MAV heading towards it, that plan is then used. Otherwise, a plan found by the trajectory planning is selected. An example of a visualization of a two-dimensional grid with information about the already scanned space and its frontiers is shown in Figure 5(b).

D. DETECTION AND ESTIMATION OF THE POSITION OF FIRES

It would be too complicated to safely create real fires inside a building. Therefore, the organizers of the competition decided to use artificial fires, hereafter referred to as fire analogues. These fire analogues are distinguishable both thermally and visually. The main body is made from plexiglass and is divided into two separate parts. The first part contains an anodized aluminium heating element with dimensions of 60 mm × 35 mm heated to 120 °C. This part is accessible through a 150 mm wide circular opening, and the task of the MAV is to spray water through this opening. The second part contains a silk flame that visually emulates flames and allows spectators of the competition to see which of the fire analogues was active. The second part is placed behind the first on the side opposite to the opening. Examples of what the fire analogues looked like during the competition are shown in Figure 11.

For effective extinguishing, it is necessary not only to detect the heating element in a thermal image, but also to detect its relative position w.r.t. the MAV in 3D. It is also necessary to obtain an estimate of the normal vector of the front plate of the fire analogue object in order to select the optimal extinguishing position for the MAV.

The heating elements are heated to 120 °C, but our thermal cameras[2] report them as being at a temperature of only approximately 70 °C. This is due to the material having an emissivity value of 0.55 [39], as opposed to the value

[2]https://terabee.b-cdn.net/wp-content/uploads/2020/05/evo-thermal_specsheet.pdf

Algorithm 3 Indoor Exploration

Input: horizontal FOV of the thermal camera ϵ

function Explore_interior(ϵ)
  $\mathcal{B}_{plan} \leftarrow [\,]$ ▷ initialize 3D grid for planning
  $\mathcal{B}_{expl} \leftarrow [\,]$ ▷ initialize 2D grid for exploration
  while not Fire_detected() **do**
    $\mathcal{R} \leftarrow$ Get_current_state_of_MAV() ▷ $\mathcal{R} = (\mathbf{x}, \mathbf{R})$
    if Received_new_data_from_sensors() **then** ▷ From the stereo camera or 2D LIDAR
      $d_{new} \leftarrow$ Get_new_data()
      $\mathcal{B}_{plan} \leftarrow$ Update_planning_buffer($\mathcal{B}_{plan}, d_{new}, \mathcal{R}$)
      $\mathcal{B}_{expl} \leftarrow$ Update_exploration_buffer($\mathcal{B}_{expl}, d_{new}, \mathcal{R}, \epsilon$)
    if Is_time_to_replan() **then**
      $f \leftarrow$ Get_nearest_frontier($\mathcal{B}_{expl}, \mathcal{R}$)
      if Is_empty(f) **then**
        return False ▷ Space has been explored without detection
      $Plan \leftarrow$ Plan_trajectory($\mathcal{B}_{plan}, \mathcal{R}, f, \epsilon$) ▷ $Plan = [\mathbf{r}_1, \eta_1, \ldots, \mathbf{r}_n, \eta_n]$
      Fly_trajectory($Plan$)
  Hover() ▷ Stop following the previous trajectory
  return True ▷ Fire detected

of 0.95 that our cameras internally use in calculating all surface temperatures. This means that the contrast between the heating elements and their surroundings in the thermal image is less than would be expected based on the temperatures alone (see Figure 6 on the left). This, however, was not a significant issue indoors, where the environment does not contain objects of such a temperature, and the viewing distances are short due to the limited size of the interior.

We could therefore merely detect the fire sources in the thermal image by binarizing the image with a fixed threshold lower than the typically measured temperature of the heating elements (Figure 6, right). In order to avoid detecting a uniformly heated background, such as a sun-heated wall, we additionally validated the detections using a differential image produced by the Laplacian image operator to check if the detected object is significantly hotter than its surroundings (Figure 6, center). It should be noted that the plexiglass casing of the fire analogues is not transparent to the infrared radiation used by our thermal cameras. This means that the observation angles, w.r.t. the front wall from which they can be seen, are limited. The observation has to be made and the extinguishing has to be done from a position as close to perpendicular with the wall. Since the radius of the circular front opening in the plexiglass casing is 7.5 cm and the heating element is positioned *approx.* 6 cm inwards from the front plate, the heating element can be seen from at most 57° from the perpendicular position. However, this is an extreme where we would only have a line of sight to the very edge of the heating element, which may not even appear on the camera. It is still desirable to maintain perpendicular alignment for extinguishing, since this maximizes the image area of the heating element, minimizes the influence of the parallax between the heating element and the front plate to which we measure the distance, and additionally maximizes the robustness of the correct aiming w.r.t. drifting in an arbitrary direction.

We assume that the thermal cameras have the properties of pinhole cameras and derive their focal distance from the pixel resolution w per side and their FOV ϵ per side. This assumption is based on the minuscule size of the cameras, the small FOV, the relative rarity, and the high cost of infrared compatible lenses. More precise calibration than this is impractically complicated to achieve, due to the low resolution of the cameras.

When a contour of compliant pixels is detected in the thermal image, we calculate the average x-y image coordinates of these pixels. These coordinates are converted to direction vectors using the assumed camera model:

$$\mathbf{v_t} = \begin{bmatrix} v_{t_x} \\ v_{t_y} \\ v_{t_z} \end{bmatrix} = \begin{bmatrix} 1/f & 0 & -((w-1)/2)/f \\ 0 & 1/f & -((w-1)/2)/f \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}, \quad (2)$$

where

$$f = \frac{(w/2)}{\tan(\epsilon/2)}. \quad (3)$$

For convenience in subsequent operations, vector $\mathbf{v_t}$ is normalized and is transformed into a coordinate frame centered in the optical center of the camera, with the x-axis pointing forwards, the y-axis to the left, and the z-axis upwards. This coordinate frame is called the thermal base frame. The transformed vector is denoted as $\hat{\mathbf{v}}_\mathbf{f}$.

For extinguishing action carried out by MAVs, we also need a distance estimate. This is achieved by combining the direction vectors with a surface shape measurement source. We use the 2D LIDAR sensor to estimate the outline of the fire analogues in front of the camera in the form of a set of

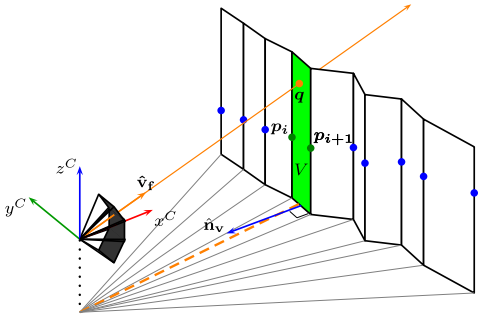


FIGURE 7. Illustration of how the fire is localized.

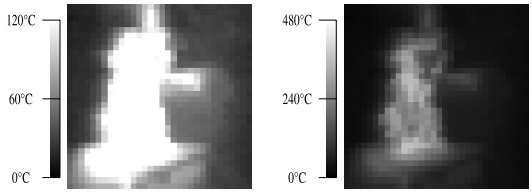


FIGURE 8. An example of the view of a real fire with the thermal camera used in the competition at two different thermal ranges. Note the high contrast of the fire compared to the background, in addition to the large size of the fires in the image. In the range of temperatures used in the competition, the fire itself is completely saturated in the image. Detecting and targeting such objects is significantly easier to achieve than when fire analogues are used.

coplanar 3D points $\mathcal{P}_{lidar}$ ordered by their angle from the sensors (blue and green points in Figure 7). This works by presuming that the entire scene is a vertical extrusion object with equal cross-sections at all heights. We can then represent the scene as a set of vertical planes passing through pairs of 3D points adjacent in the sense of angle (white and green vertical planes in Figure 7). We select a pair of points $\boldsymbol{p}_i$ and $\boldsymbol{p}_{i+1}$ from $\mathcal{P}_{lidar}$, s.t.

$$\text{atan2}(p_{i_y}, p_{i_x}) > \text{atan2}(v_{f_y}, v_{f_x}), \tag{4}$$

$$\text{atan2}(p_{i+1_y}, p_{i+1_x}) < \text{atan2}(v_{f_y}, v_{f_x}). \tag{5}$$

The 3D position of the estimated target is then obtained by calculating the intersection of the optical line (the orange line in Figure 7) with the vertical plane V (the green plane in Figure 7) passing through the selected pair of points. Plane V is defined by point $\boldsymbol{p_i}$ and normal $\mathbf{n_v}$:

$$\mathbf{n_v} = \begin{bmatrix} -(p_{i+1_y} - p_{i_y}) \\ p_{i+1_x} - p_{i_x} \\ 0 \end{bmatrix}. \tag{6}$$

The intersection point $\boldsymbol{q}$ is then calculated as:

$$\boldsymbol{q} = \hat{\mathbf{v}}_\mathbf{f} \cdot t, \tag{7}$$

where t is obtained using the normalized vector of the surface normal $\hat{\mathbf{n}}_\mathbf{v}$ as:

$$t = \frac{\hat{\mathbf{n}}_\mathbf{v} \cdot \boldsymbol{p}_i}{\hat{\mathbf{n}}_\mathbf{v} \cdot \hat{\mathbf{v}}_\mathbf{f}}. \tag{8}$$

The estimate of the surface normal $\hat{\mathbf{n}}_\mathbf{v}$ and the intersection point $\boldsymbol{q}$ are used to steer the MAV into perpendicular position for extinguishing $\boldsymbol{s}$, defined as

$$\boldsymbol{s} = \boldsymbol{q} + r_d \cdot \hat{\mathbf{n}}_\mathbf{v}, \tag{9}$$

where r_d is the desired extinguishing distance of 1.5 m.

We did not consider a single estimate of the 3D position of the heat source to be sufficient. Instead, we implemented a Kalman filter that stores multiple measurements as an array of states and refines each state using new measurements. The state vector of the Kalman filter used here is

$$\mathbf{x}_i = \left[c_{i_x}, c_{i_y}, c_{i_z}, \psi_i\right]^T, \tag{10}$$

where c_{i_x}, c_{i_y}, and c_{i_z} are the coordinates of the fire in the world coordinate frame. The ψ_i represents the azimuth of the surface normal for that fire. The filtering mechanism stores multiple such state vectors, corresponding to multiple different detected fires. We update a specific state vector $\mathbf{x}_i$ using a new estimate of the fire position $\boldsymbol{q}$ and normal $\mathbf{n_v}$, if $\boldsymbol{q}$ is closer in the world frame than 1 m to $\left[c_{i_x}, c_{i_y}, c_{i_z}\right]^T$ and at the same time the horizontal component of $\mathbf{n_v}$ is closer than 90° to ψ_i. If no such state is found, a new state is initialized based on the current estimate. As is typical of the Kalman filter, the state covariances grow in time to reflect loss of knowledge without observation and, in our case, additionally discard any measurements or even states that exhibit erroneous properties or states that have not been updated with a measurement for the past 10 s. To account for random errors, a state $\mathbf{x}_i$ is only used in fire extinguishing if it has been associated with at least 10 measurements.

It should be noted that the method for thermal detection and localization of fire analogues used here is significantly more complicated than would be realistically required in extinguishing real-world fires. This is because fires that pose a real danger are significantly hotter than the heating elements of the fire analogues used in the competition, and dangerous fires do not appear colder than they actually are in thermal cameras (see Figure 8 for an example). Additionally, these fires are significantly larger objects, and extinguishing them would not require just spraying into a very narrow opening. In real fire extinguishing, it is desirable to aim at the hottest areas detected in the fire. A more dispersed water stream would also be advantageous, as it would be more likely to hit the fire, even with less precise target localization, and it would extinguish the fire more effectively.

E. FIRE EXTINGUISHING

Upon obtaining the first validated fire detection state in the Kalman filter array, the MAV is sent to a position $\boldsymbol{s}$ 1.5 m in front of the given target $\boldsymbol{q}$ along the estimated normal $\mathbf{n_v}$. As the MAV flies there, its estimate of the target position and the surface normal improves as it obtains new detections from better viewpoints. When it reaches the position, control is handed over to the fire-extinguishing subsystem (the state labeled *Extinguish fire* in Figure 10(d)).

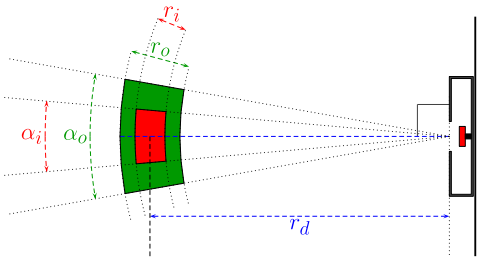


FIGURE 9. Steering hysteresis for fire extinguishing, viewed from above.

In this state, the MAV is steered towards position s. However, continuous positional stabilization of the MAV towards this exact position may lead to rapid tilting, due to movement oscillations during attempts to correct the current position, especially in face of potential fire estimation errors and disturbances such as air currents. This causes the direction of the water stream to be significantly unstable, since the water nozzle does not have active stabilization due to weight restrictions.

To mitigate this effect, we have included a hysteresis to horizontal steering through two ranges of angle and distance offsets (see Figure 9). The angle ranges were defined as limits to the angles formed by the surface normal $\mathbf{n_v}$ and the line connecting the center of the fire analogue with the MAV - the inner range α_i was set to $\pm 5°$, and the outer range α_o was set to $\pm 10°$. The distance ranges are offsets from the desired extinguishing distance of 1.5 m. These were set to ± 0.075 m for the inner range r_i, and to ± 0.15 m for the outer range r_o. Once the MAV has reached the inner ranges (the red zone in Figure 9), the xy-coordinates are not corrected, irrespective of disturbances. The system can only correct its altitude and its heading as changing these does not generate tilt of the MAV. The MAV thus tends to drift or "float". It is necessary to correct the heading continuously, otherwise the drifting would affect the aim of the MAV. Water spraying is only activated when the MAV is in this drifting state. The z-coordinate and the heading are controlled either to spray at the directly observed heating element of the fire analogue, or - if it is not visible *e.g.* due to being cooled down by the water - to spray at its estimated position from the Kalman filter. If the target is directly observed, the aiming is more responsive to disturbances. However, if the aiming relies on the filter, the precision is lowered. The MAV is only allowed to correct its xy-coordinates again when it has been moved outside the outer ranges (the green zone in Figure 9), at which point water spraying is disabled.

F. HIGH-LEVEL BEHAVIOR CONTROL

The complete behavior structure of the proposed system is constructed as a hierarchical state machine, which is used for interconnecting all the subsystems. The state machine was designed for robustness of the entire code structure by resolving the remaining few subsystem failure cases due to wrong sequential and concurrent operations. The hierarchical state machine is implemented using the Flexbe library [40], and it is fully integrated into the designed ROS framework.

In Figure 10, the internal states of the state machines are visualized as single-outline rectangles, and the nested lower-level state machines are visualized as double-outline rectangles. Transitions between two states and from one state machine to a lower-level state machine are marked by arrows with labels of outcomes describing the given transition. Dotted terminal states represent the transition that is called after returning to a higher level state machine. A landing event is called whenever any state produces an outcome that means that the MAV cannot continue its mission. Unfortunately, there is no information available for the MAV to recognize whether the amount of the sprayed liquid was sufficient to extinguish the fire. Therefore, whenever the MAV lands, the operators can see whether or not the mission was successful by the state of the water bag.

The diagram of the main state machine is visualized in Figure 10(a). In the first step, the correct performance of all key parts of the system is checked. When every component is verified to be operational, an automatic takeoff is called. Once the MAV is in the air, the mission commences. The mission is divided into two parts: the outdoor phase and the indoor phase. The outdoor phase is the part of the mission where the MAV is outside the building. The indoor phase is when the MAV is inside the building. At the end of the mission (a window or a fire has not been found, or a fire has been successfully extinguished), the MAV flies back to the home position and lands.

The outdoor phase (Figure 10(c)) starts by flying to the known GNSS position of the building. This position must be a position from which the MAV is capable of detecting the building. A common problem with navigation using standard GNSS is its precision, which depends on the quality of the satellite signal. GNSS satellites broadcast their signals in space, but what we receive depends on additional factors, such as signal blockage and atmospheric conditions. Therefore, a safe position in front of the building may drift into the building. For this reason, the MAV uses scans provided by 2D LIDAR during the flight to facilitate navigation around the building where the GNSS quality is degraded by the building blocking the direct line of sight of some satellites. These scans provide planar information in 360° around the MAV and are used in a virtual bumper. The virtual bumper is a system that prevents the MAV from following a plan that would lead it to go closer than the predefined safe distance from the building. If the target position is inside the building, the MAV will stop at a position within a safe distance from the building and closest to the target position.

After it reaches a safe position near the building, the MAV starts flying alongside the building at a predefined distance with a heading towards the building, and begins the window detection mechanism. Whenever a window is located, the MAV stops flying alongside the building and flies in front

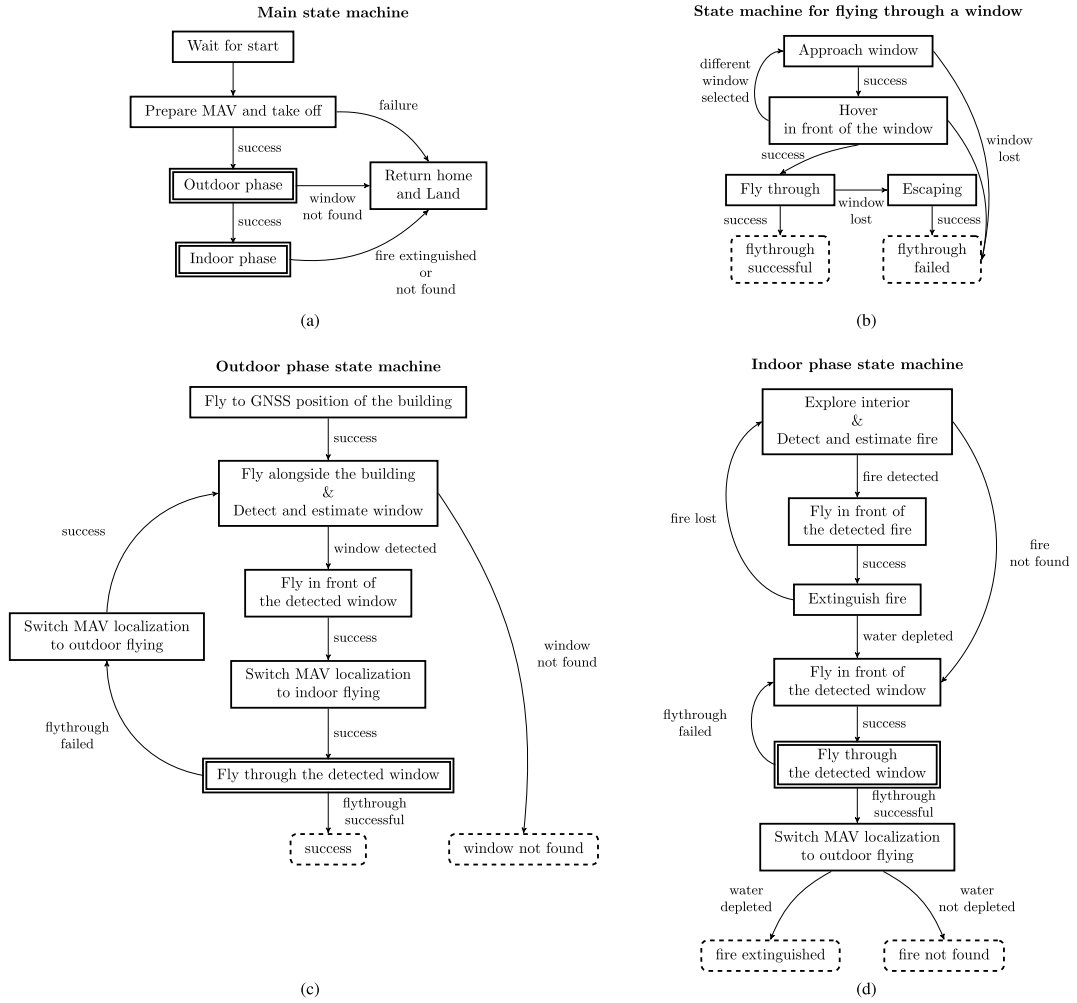


FIGURE 10. Diagrams of the main parts of the proposed system.

of the window to distance of 2 m from its center. Once this position is reached, the localization of the MAV is switched to indoor flying mode (LIDAR-based odometry is used in the controller feedback), and an attempt is made to fly through the window. If the attempt is successful, the MAV is inside the building and the outdoor phase is considered successfully finished. In the case of the opposite result, the MAV restarts the search for an open window. After circumnavigating the building without detecting a window, the outdoor phase ends, and the result is registered as ''window not found''. The MAV is allowed to detect the same window again and to attempt to fly through it. This strategy is motivated by the knowledge that only a single window was to be opened on each floor during the competition trial. The attempts can be repeated until the maximum allowed flight time is reached. After reaching this time, the MAV automatically lands.

The steps for flying through the selected window are shown in Figure 10(b). The procedure utilizes window estimates produced by algorithms described in section II-B. First, the MAV flies to a position in front of the window while continuously facing the center of the window. The MAV then hovers in front of the center of the window to stabilize itself before the actual flythrough. The flythrough maneuver is then initialized and the state machine waits for an up-to-date window estimate corrected by new detections. After the window estimate has been updated, the MAV flies through the center of the

FIGURE 11. Photos of the MBZIRC 2020 Fire Challenge area. The photo on the left shows the tall structure simulating a building, the photo in the middle shows front view of the fire analogues (this unit was turned on) and the photo on the right shows side view of the fire analogues (this unit was turned off).

window to a goal position at a predefined distance behind the window while maintaining a constant altitude. If the window estimate is lost while the flythrough is in progress and the MAV is still outside the building, the state machine switches to the Escaping state and the MAV returns to its original hovering position in front of the window.

The indoor phase (Figure 10(d)) contains the final parts – localization and extinguishing of the fire. Localization is done by using the exploration method, which is described in section II-C, and the detection system, which is described in section II-D. Once the fire is detected, the MAV flies in front of it and begins extinguishing (section II-E). If the fire target is not lost, the MAV depletes all the water that it is carrying during the extinguishing maneuver. There is no feedback that provides information as to whether the extinguishing has been completed. The extinguishing is therefore declared completed once all the water is depleted. In the case that the fire is lost, the MAV starts exploring again. After depleting the extinguishing agent, the MAV flies back in front of the window that it entered through and tries to fly back outside the building. An attempt to leave the building using the same window is also performed if the exploration finishes without successful localization of the fire. When the MAV is outside, the localization of the MAV is switched to outdoor flying again and MAV flies back to land on the starting position.

In the case of a real firefighting scenario inside a building, the proposed system can be used in almost the same structure as presented here. The only modification is that the process of searching for an open window can be accelerated by directly specifying the approximate GNSS position of the window.

III. EXPERIMENTAL RESULTS

A. SIMULATIONS

To be able to experimentally verify the entire firefighting mission, we modeled the MBZIRC 2020 scenario in the Gazebo robotic simulator. The interior of the building was updated during the competition to correspond with the interior of the real building, as observed during the rehearsals (see photos from the competition in Figure 11). The Fire Challenge arena is approximately 50 m × 60 m in dimensions and contains a tall structure (18 m in height) simulating a building. The interior of each floor of the building contains two fire analogues and only one per floor is activated during the trial. Each floor of the building contains eight 2 m × 2 m windows. Only one of them is open and can be used as an access point to enter

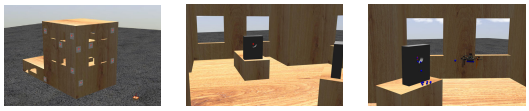


FIGURE 12. Snapshots from the simulation developed for the Fire Challenge of the MBZIRC 2020 competition.

the floor. Snapshots from the simulated scenario are shown in Figure 12.

The behavior of the proposed system can be simulated completely, including the outdoor and indoor flying, window detection, fire detection and also fire extinguishing (see the right image in Figure 12). Numerous simulations were conducted with different settings of the system parameters during the preparations for the competition. The results of the system in the final form after the competition obtained for the evaluation of the system for this paper are shown in Table 1. The goal of each run of the simulation was to extinguish an artificial fire on the first floor of the building. For testing purposes and according to the rules, one of the fire analogues (windows) was randomly selected and was turned on (opened). The position where the MAV started was the same each time for each run of the simulation. Three performance criteria may be considered for an evaluation of the task under discussion in this paper: reliability, total mission time, and minimal distance from the obstacles. The results show that the mission can be completed within 7 min. However, the fire analogue was detected and extinguished only in 80 % of the cases, due to problematic properties of the fire analogues. The fire analogues are visible in the thermal images only under a viewing angle of at most 57° from the position perpendicular to the fire analogue. The proposed exploration method does not consider the angle under which the particular surface in the scene is observed. Therefore with the fire analogues, the system can consider the surface as already scanned even though the fire analogue was not detected, because the surface was scanned under an angle from which the heated element could not be detected. This is a specific property of the fire analogues used in the competition, and it will not prevent successful detection of real fires. During these simulations, it was successfully verified that the MAV did not come closer to the obstacles than 0.7 m, which was the minimum obstacle distance set for the indoor motion planning algorithm.

B. REAL-WORLD VERIFICATION

1) PLATFORM DESCRIPTION

Our team participated in all challenges of the MBZIRC 2020 competition. To allow reusability of the system and the spare parts, we decided to select a base MAV platform that can be used in all challenges, with possible modifications to the sensors and actuators. The proposed firefighting MAV platform with the complete sensory equipment is shown in Figure 13.

The selected base platform is created mostly from commercially available off-the-shelf components and 3D printed

TABLE 1. Table with results from 10 runs of the simulated MBZIRC 2020 competition scenario. The MAV returned back to the starting position each time. However, the fire analogue was detected and was extinguished only in 8 of the cases, due to problematic properties of the fire analogues.

	1	2	3	4	5	6	7	8	9	10
Total time [s]	336.3	368.2	222.3	226.4	218.5	223.1	258.9	165.3	246.9	321.6
Min. distance [m]	0.83	0.73	0.74	0.83	0.78	0.78	0.8	0.87	0.74	0.77
Mission successful?	✓	×	×	✓	✓	✓	✓	✓	✓	✓

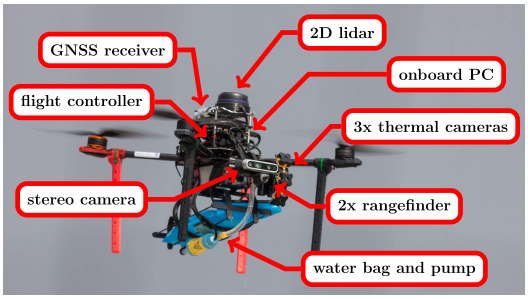


FIGURE 13. Description of the components of the deployed MAV platform for indoor fire extinguishing.

parts. The platform is built from the *Tarot T650* quadrotor frame, the *PixHawk 4* flight controller,[3] and an *Intel NUC* onboard computer. This frame satisfies the size limitations set for the competition and also for the real applications (the diagonal dimension without propellers is 650 mm), and provides the payload capacity that is necessary for carrying additional sensors and fire-extinguishing equipment. The onboard computer is Intel NUC8i7BEH,[4] which contains Intel i7-8559U CPU and 8 GB of RAM, and runs the Ubuntu 18.04 LTS operating system and Robot Operating System (ROS) [41] Melodic middleware. In addition, the MAV is equipped with *RPLIDAR A3*,[5] which is a 360° 2D laser range scanner that can be used for both indoor and outdoor applications. This sensor provides 16000 samples per second and can detect obstacles within a 25 m radius, depending on the setting of the sensor. For the stereo camera, we use the *Realsense D435* camera,[6] which has FOV (H × V × D) 87°±3° × 58°±1° × 95°±3° and a range of up to 10 m. Fire detection is done using a set of three *TeraRanger Evo Thermal 33*[7] thermal cameras. This thermal camera is cheap, small, and compact (only 12 g), which is very important in this case of a limited payload. However, the camera has small resolution of 32 × 32 pixels and FOV of 33° in both dimensions, and requires a set of three of these sensors onboard the MAV to cover the vertical space in front of the MAV sufficiently for this application. The cameras are arranged vertically, with s.t. one pointing forward and the two others above and below it, oriented 30° upwards and downwards from the first camera (see Figure 13). The MAV is further equipped with two *Garmin LIDAR-Lite v3*[8] laser rangefinders.

To extinguish fires, the MAV is equipped with a water bag and a pump.[9] The capacity of the bag was limited to 1 L of the fire-extinguishing agent (water in the case of the competition) to maintain higher maneuverability of the system. This maneuverability is vital for flight in an environment, such as the inside of a building, where strong air currents and various obstacles can be encountered. The pump drives the water through a nozzle with a diameter of 4 mm and can fully deplete the bag within 25 s. The nozzle is rigidly attached to the MAV frame, and is oriented to the front with the spraying tip located 2 cm below and 2 cm in front of it. As has already been mentioned, this nozzle is not actuated, since a servomechanism of this type would significantly increase the weight of the MAV.

2) EXPERIMENTS

The key parts of the proposed system were thoroughly tested in demanding outdoor conditions in the desert near Abu Dhabi in the United Arab Emirates. This environment was selected to emulate the conditions around buildings and other conditions set for the competition itself (mainly sudden wind gusts, strong sunlight and dust), while providing a safe field for system tuning and experimental verification. Repeated experimental verification of the key parts of the proposed system was necessary in order to prepare for phenomena that are difficult to simulate, and also to discover issues related to the hardware that was deployed. One issue that emerged was the influence of sensors connected using USB 3.x., such as *Realsense D435*, on the precision of GNSS. The precision of GNSS localization can be severely decreased by the influence of components transmitting via cable at frequencies close to those used by GNSS. See sheet[10] for a detailed description of USB 3 frequency interference. It was therefore necessary

[3] https://github.com/PX4/px4_user_guide/raw/master/assets/flight_controller/pixhawk4/pixhawk4_technical_data_sheet.pdf
[4] https://www.intel.com/content/dam/support/us/en/documents/mini-pcs/NUC8i3BE_NUC8i5BE_NUC8i7BE_TechProdSpec.pdf
[5] https://www.generationrobots.com/media/LD310_SLAMTEC_rplidar_datasheet_A3M1_v1.0_en.pdf
[6] https://www.intelrealsense.com/wp-content/uploads/2020/06/Intel-RealSense-D400-Series-Datasheet-June-2020.pdf
[7] https://terabee.b-cdn.net/wp-content/uploads/2020/05/evo-thermal_specsheet.pdf
[8] http://static.garmin.com/pumac/LIDAR_Lite_v3_Operation_Manual_and_Technical_Specifications.pdf
[9] https://www.comet-pumpen.de/fileadmin/pdf/pumpen_datenblaetter/24v/Datenblatt_VIP-PLUS_24V_1435.88.00.pdf
[10] https://www.intel.com/content/www/us/en/products/docs/io/universal-serial-bus/usb3-frequency-interference-paper.html

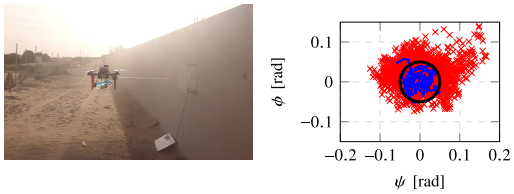


FIGURE 14. Experimental verification of fire extinguishing in the desert near Abu Dhabi, United Arab Emirates. The image on the left shows a photo from this experiment. The image on the right displays the plot of the changing offsets of the nozzle direction from the estimated fire analogue direction during the extinguishing procedure. Angles ϕ and ψ denote vertical (pitch) and horizontal (yaw) angular offsets. The blue line denotes when the water pump was activated. The black circle denotes the range of offsets corresponding to the area of the opening in the plexiglass viewed from the desired extinguishing distance of 1.5 m. Note that most of the time, the water only sprinkled when the nozzle was aiming into the opening. However, the real world dynamics of the water stream caused more spillage than the aim itself implies. A video of this experiment is on YouTube[11].

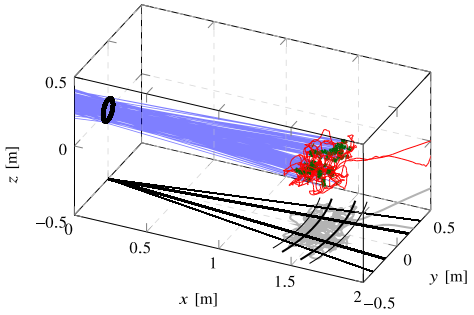


FIGURE 15. Plot of the MAV trajectory during the extinguishing experiment displayed in Figure 14. The red line denotes the trajectory itself relative to the fire analogue. The green points are specific positions from which the MAV activated its water pump. Spraying lines parallel to the direction of the water nozzle in the spraying positions are projected in blue. Note that 93% of these spraying lines pass through the black circle denoting the opening in the plexiglass of the fire analogue. The trajectory is projected to the *xy*-plane as a shadow, for better clarity. Also shown in the *xy*-plane are the hysteresis ranges described in Figure 9.

to shield the receiver of the GNSS signal. Another issue that we discovered was the necessity to calibrate the fire-extinguishing device to hit the fire detected by the thermal cameras precisely for each MAV. Otherwise, the ejected water would not precisely hit the opening in the fire analogues. In fact, the direction of the water stream was diverted downwards by the pressure generated by the active propellers. Although precise placement of the fire-extinguishing agent is also important consideration in real firefighting, the small size of the opening in the MBZIRC 2020 fire analogues presented a much more difficult challenge than a firefighting MAV would face in a real fire.

FIRE DETECTION AND EXTINGUISHING

The first experiment presented here was focused on fire detection in conjunction with autonomous fire extinguishing (discussed in section II-D and section II-E). The initial goal of this experiment was to detect the fire analogues. After successful detection, the MAV moves in front of the fire at a distance of 1.5 m while heading towards the opening in the center of the fire analogue, and then it initiates the autonomous fire extinguishing. Whenever the MAV points the nozzle towards the opening at the correct relative distance, the water pump is activated (see Figure 14 and Figure 15). The experiment shown in Figure 14 and 15 was carried out in the latest stage of system development prior to the competition, representing the final state of the fire-extinguishing subsystem. A video showing this experiment is available on YouTube.[11] As the data shows, at least in terms of position and heading, the MAV approached the desired extinguishing position w.r.t. the fire analogue and deployed water into the small opening. In this experiment, for 93 % of the time when the water pump was activated, the water nozzle was aiming correctly at the opening. The remaining 7 % was affected by the delay until the pump turned off successfully. This shows the accuracy of the fire detection and localization system. It should be noted that some of the deployed water was lost due to various effects such as dispersion, the momentum of the liquid in the spraying system, surface tension within the water stream, bouncing off from the back plate of the fire analogue, stronger ballistic curvature when the pump is being activated or deactivated, and evaporation from the heating element. Note also that there were numerous interruptions in the correctly-aimed water spraying. These interruptions were caused by loss of the target by the thermal cameras due to the heating element being temporarily cooled down by the deployed water (this was a special property of the MBZIRC 2020 fire analogue, not of a real fire). For this implementation, we decided that it was a better strategy to let the target heat up again and to invest additional time instead of continuously spraying merely based on the "remembered" position of the target. Such estimation without new measurements drifts from the real position, and the limited carrying capacity of an MAV makes it a priority to be economical with the extinguishing agent. With real hazardous fires, such losses of vision will only occur after the extinguishing has been successful, so it would not be necessary to interrupt the water stream.

INDOOR MOTION PLANNING AND EXPLORATION

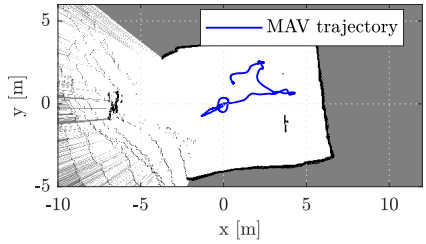
Another experimentally verified subsystem was indoor motion planning and exploration (discussed in section II-C). The goal of the experiment was to completely explore the space inside a room with obstacles consisting of poles holding the structure and wooden artificial obstacles. An example layout of the obstacles inside the room is shown in Figure 16(a). A visualization of data from the experiment in this setup is shown in Figure 16(b). Figure 17 shows a visualization of the MAV trajectory and minimal obstacle distance progression from one of the performed flights. During

[11] https://youtu.be/9bkvfi5uHK4

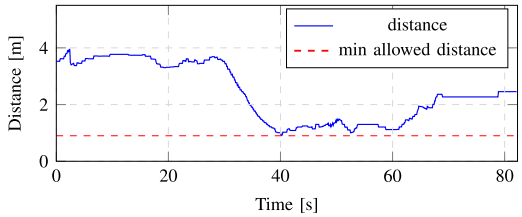
(a) Side view from an external camera.

(b) Data visualization.

FIGURE 16. Experimental verification of the indoor motion planning and exploration techniques in the desert near Abu Dhabi, United Arab Emirates before MBZIRC 2020. In the image on the right, the red blocks denote obstacles, the green blocks show the already scanned space, and the yellow blocks denote frontiers for exploration. A video from this experiment is on YouTube[12].



(a) MAV trajectory with a map produced by Hector SLAM is drawn in the background.



(b) Distance of the MAV from obstacles during the flight.

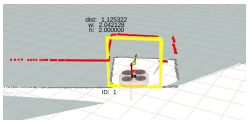
FIGURE 17. Data from one of the flights from the indoor motion planning and exploration experiment displayed in Figure 16.

the flights, the MAV had not come closer than the specified minimal distance of 0.9 m to the observed obstacles, and had successfully explored the space inside the room. A video showing one of these experiments is available on YouTube.[12]

WINDOW DETECTION AND THE OUTDOOR-INDOOR TRANSITION

Another experiment in the desert was aimed at verifying the correct performance of outdoor wall following, window detection, flight through the detected window, and switching between indoor and outdoor modes of localization. For this purpose, a wooden room was constructed next to a long wall. The room was approximately 2.5 m × 3 m × 2 m in size and contained an entrance 2 m × 2 m in size. The size of the room did not match the MBZIRC 2020 specification but it was suitable for testing these particular parts of the system. Figure 19(a) displays the trajectory of the MAV during the experiment. The MAV started 2.5 m from the wall and autonomously detected the wall as the closest object seen by the 2D LIDAR, then followed the wall at a distance of 2 m while simultaneously searching for a window. During the outdoor flight, the MAV was localized using GNSS. The window was detected using a combination of 2D LIDAR data and a priori information about its size and altitude. When the window was detected, the MAV approached the window, after which the localization switched to indoor mode (using 2D LIDAR-based Hector SLAM), and the MAV flew inside. The MAV then turned around inside the building, flew back outside, and the localization mode switched back to GNSS. Finally, the MAV returned back to its starting position and landed. The whole experiment, along with a visualization of the sensor data, can be seen in a video on YouTube.[13] Figure 18 shows the MAV in front of the building along with a visualization of the sensor data and the detected window. Figure 19(b) contains a plot of the total control error (defined as the 3D Euclidean distance between the current reference and the MAV position) from the entire flight. This graph shows that the switch between the two different localization systems was smooth and did not impact the control of the MAV. The average control error during the flight was 0.14 m.

(a) MAV in front of the mock-up building.



(b) Visualization of sensor data.

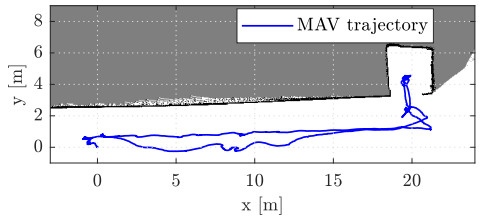
FIGURE 18. Images from experimental verification of the correct performance of outdoor wall-following, window detection, flight through the detected window, and switching between indoor and outdoor modes of localization. 18(a) shows the MAV before entering a mock-up building in the desert and 18(b) displays a particular visualization of the onboard sensor data. The visualization includes the MAV position, LIDAR data shown as red squares, the detected window as a yellow rectangle, and a map produced by the Hector SLAM algorithm. A video from this experiment is on YouTube[13].

COMPLETE SYSTEM VERIFICATION

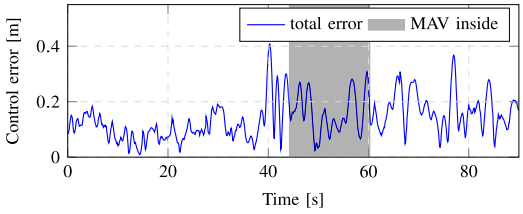
The complete system was tested in a mock-up of the competition building set up in the Czech Republic. The mock-up is 5 m × 5 m, with 2 floors totaling 5 m in height, with windows 1.85 m × 1 m in dimensions (see Figure 20). The size of the windows matches the first specification for the MBZIRC 2020 competition. This specification was changed later, and the windows are smaller than in the competition. This made the flight through the window more challenging than was necessary for the competition, but it verified the performance and the robustness of the system for real-world deployment. In the experiment presented here, the MAV began next to the mock-up, autonomously detected the wall as the closest object seen by 2D LIDAR, and then followed

[12]https://youtu.be/9LTf6PG4ijc

[13]https://youtu.be/aCKUjbJ2Mxs



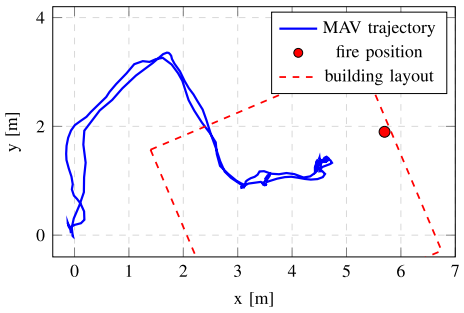
(a) MAV trajectory with a map produced by Hector SLAM drawn in the background.



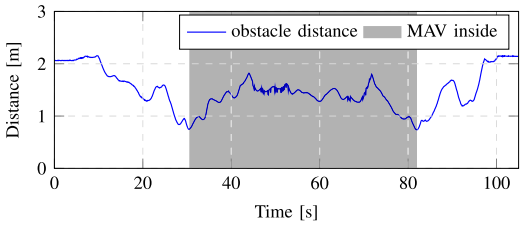
(b) Control error plot.

FIGURE 19. Data from the outdoor-indoor transition experiment displayed in Figure 18.

FIGURE 20. Experimental verification of the proposed system on a mock-up of the building set up in the Czech Republic. The image on the left shows the MAV outside the mock-up during its search for an open window. The image on the right displays the MAV while the water is being sprayed on the fire analogue inside the mock-up. A video of this experiment is on YouTube[14].



(a) Trajectory traveled by the MAV projected to the *xy*-plane.



(b) Closest obstacle distance measured by the 2D LIDAR.

FIGURE 21. Data from the complete system verification experiment presented in Figure 20.

the wall at a distance of 1.5 m while simultaneously searching for a window. The window was detected using a combination of data from 2D LIDAR and from the stereo camera. After successful detection, the MAV approached the window and flew inside. Then, the MAV started to explore the interior of the building with the goal to find the fire analogue and then to extinguish the fire. The fire analogue was later detected and all the water was depleted on it. Finally, the MAV flew out of the building, using the same window through which had flown in, and then flew back to the starting position of the mission, where it landed. The complete trajectory traveled by the MAV during this experiment is shown in Figure 21(a). Figure 21(b) shows the distance from the closest obstacles measured by 2D LIDAR during the flight. The shortest obstacle distance of the whole flight was 0.74 m, when the MAV was flying through the window. It can be seen that the MAV motion was successfully planned with an adequate safety margin throughout the flight. The whole experiment, along with a visualization of the sensor data, can be seen in a video on YouTube.[14]

IV. GOING BEYOND THE MBZIRC COMPETITION

An MAV can carry a water bag filled only with a very limited amount of water, due its limited payload capacity. Even after this entire amount has been discharged perfectly into a real fire, the fire will very likely not be extinguished. To increase the payload capability of the MAV, its size has to be changed. However, greater dimensions of the MAV will make flying through windows and inside buildings very difficult to achieve, if not even impossible. Therefore, in the most cases the fire-extinguishing approach, with the water spray, is not the optimal solution.

Based on our results in the competition, the proposed autonomous system was selected to be the core of an industrial firefighting MAV system using fire-extinguishing capsules.[15] This system makes it necessary to hit the fire directly, meaning that it requires reliable techniques for locating, approaching, and aiming precisely at fires. These techniques are being adapted from the work presented here, combined with a throwing mechanism able to place the active fire-fighting capsules quickly and precisely.[16] A prototype of a complex industrial platform is shown in Figure 22.

[14]https://youtu.be/a-VsVQcMLuQ
[15]http://www.fire-defender.com/en/bonpet-3/1465-2
[16]http://mrs.felk.cvut.cz/projects/dofec

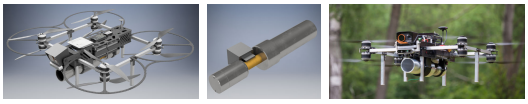


FIGURE 22. A new platform under development that uses fire-extinguishing capsules as projectiles. The first two photos on the right show 3D render of the platform and launcher design, photo on the left shows constructed platform.

V. CONCLUSION

In this work, we have presented a complex system developed for fully autonomous extinguishing of fires inside a building using an MAV system. The challenges include control and estimation of the MAV state, interior motion planning and exploration, window and fire detection and position estimation, and fire extinguishing. One of the main contributions of the system lies in precise control for flying through relatively small windows, and also for precise spraying of fire-extinguishing agent into a small opening representing the fire, using multiple sensory data to increase reliability. For this, we have presented a novel multi-layer control pipeline that further enables precise localization and stabilization in an open space around a building, inside rooms with obstacles, and also with a smooth transition between these two environments (with GNSS and GNSS-denied). This smooth transition is another contribution that motivated the MBZIRC 2020 committee to design this demanding challenge. The paper contains the system performance presented in simulations and field tests in various demanding real-world conditions. The system was developed as part of a solution for the firefighting mission in the MBZIRC 2020 competition, and it helped our team to achieve first place in the Grand Challenge of this competition among the best universities in aerial robotics worldwide.[17] Although the system was developed specifically for this competition, the solution presented here has led to an industrial solution that is now under development. This solution will be focused on real-world firefighting, in which autonomous drones will deploy fire-extinguishing capsules.

REFERENCES

[1] H. Shakhatreh, A. H. Sawalmeh, A. Al-Fuqaha, Z. Dou, E. Almaita, I. Khalil, N. S. Othman, A. Khreishah, and M. Guizani, "Unmanned aerial vehicles (UAVs): A survey on civil applications and key research challenges," *IEEE Access*, vol. 7, pp. 48572–48634, 2019.

[2] S. Verykokou, C. Ioannidis, G. Athanasiou, N. Doulamis, and A. Amditis, "3D reconstruction of disaster scenes for urban search and rescue," *Multimedia Tools Appl.*, vol. 77, no. 8, pp. 9691–9717, Apr. 2018.

[3] D. Duarte, F. Nex, N. Kerle, and G. Vosselman, "Towards a more efficient detection of earthquake induced facade damages using oblique uav imagery," in *Proc. IEEE ICUAVG*, Aug. 2017, pp. 93–100.

[4] Y. Yamazaki, M. Tamaki, C. Premachandra, C. J. Perera, S. Sumathipala, and B. H. Sudantha, "Victim detection using UAV with on-board voice recognition system," in *Proc. 3rd IEEE Int. Conf. Robotic Comput. (IRC)*, Feb. 2019, pp. 555–559.

[5] M. Rahnemoonfar, R. Murphy, M. V. Miquel, D. Dobbs, and A. Adams, "Flooded area detection from UAV images based on densely connected recurrent neural networks," in *Proc. IEEE Int. Geosci. Remote Sens. Symp. (IGARSS)*, Jul. 2018, pp. 1788–1791.

[6] R. Ravichandran, D. Ghose, and K. Das, "UAV based survivor search during floods," in *Proc. Int. Conf. Unmanned Aircr. Syst. (ICUAS)*, Jun. 2019, pp. 1407–1415.

[7] C. Yuan, Z. Liu, and Y. Zhang, "UAV-based forest fire detection and tracking using image processing techniques," in *Proc. Int. Conf. Unmanned Aircr. Syst. (ICUAS)*, Jun. 2015, pp. 639–643.

[8] J. Delmerico, S. Mintchev, A. Giusti, B. Gromov, K. Melo, T. Horvat, C. Cadena, M. Hutter, A. Ijspeert, D. Floreano, L. M. Gambardella, R. Siegwart, and D. Scaramuzza, "The current state and future outlook of rescue robotics," *J. Field Robot.*, vol. 36, pp. 1–21, Oct. 2019.

[9] T. Rouček, M. Pecka, P. Čížek, T. Petříček, J. Bayer, V. Šalanský, D. Heřt, M. Petrlík, T. Báča, V. Spurný, F. Pomerleau, V. Kubelka, J. Faigl, K. Zimmermann, M. Saska, T. Svoboda, and T. Krajník, "DARPA subterranean challenge: Multi-robotic exploration of underground environments," in *Proc. IEEE MESAS*, Mar. 2020, pp. 274–290. [Online]. Available: https://link.springer.com/chapter/10.1007/978-3-030-43890-6_22

[10] F. Mascarich, S. Khattak, C. Papachristos, and K. Alexis, "A multi-modal mapping unit for autonomous exploration and mapping of underground tunnels," in *Proc. IEEE Aerosp. Conf.*, Mar. 2018, pp. 1–7.

[11] M. Petrlik, T. Baca, D. Hert, M. Vrba, T. Krajnik, and M. Saska, "A robust UAV system for operations in a constrained environment," *IEEE Robot. Autom. Lett.*, vol. 5, no. 2, pp. 2169–2176, Apr. 2020.

[12] L. Shi, X. Wang, T. Zhang, C. Hu, K. Luo, and B. Bai, "Hazardous gas detection four-rotor UAV system development," in *Proc. IEEE Int. Conf. Mechatronics Autom.*, Aug. 2016, pp. 2461–2465.

[13] Z. Fu, Y. Chen, Y. Ding, and D. He, "Pollution source localization based on multi-UAV cooperative communication," *IEEE Access*, vol. 7, pp. 29304–29312, 2019.

[14] P. Stibinger, T. Baca, and M. Saska, "Localization of ionizing radiation sources by cooperating micro aerial vehicles with pixel detectors in real-time," *IEEE Robot. Autom. Lett.*, vol. 5, no. 2, pp. 3634–3641, Apr. 2020.

[15] C. Corrado and K. Panetta, "Data fusion and unmanned aerial vehicles (UAVs) for first responders," in *Proc. IEEE Int. Symp. Technol. Homeland Secur. (HST)*, Apr. 2017, pp. 1–6.

[16] L. Merino, F. Caballero, J. R. Martínez-de-Dios, I. Maza, and A. Ollero, "An unmanned aircraft system for automatic forest fire monitoring and measurement," *J. Intell. Robotic Syst.*, vol. 65, nos. 1–4, pp. 533–548, Jan. 2012.

[17] A. Restas, "Forest fire management supporting by UAV based air reconnaissance results of Szendro fire department, Hungary," in *Proc. 1st Int. Symp. Environ. Identities Medit. Area*, Jul. 2006, pp. 73–77.

[18] A. Viguria, I. Maza, and A. Ollero, "Distributed service-based cooperation in aerial/ground robot teams applied to fire detection and extinguishing missions," *Adv. Robot.*, vol. 24, nos. 1–2, pp. 1–23, Jan. 2010.

[19] R. Chen, H. Cao, H. Cheng, and J. Xie, "Study on urban emergency fire-fighting flying robots based on UAV," in *Proc. IEEE 4th Adv. Inf. Technol., Electron. Autom. Control Conf. (IAEAC)*, Dec. 2019, pp. 1890–1893.

[20] H. Qin, J. Q. Cui, J. Li, Y. Bi, M. Lan, M. Shan, W. Liu, K. Wang, F. Lin, Y. F. Zhang, and B. M. Chen, "Design and implementation of an unmanned aerial vehicle for autonomous firefighting missions," in *Proc. 12th IEEE Int. Conf. Control Autom. (ICCA)*, Jun. 2016, pp. 62–67.

[21] P. Pecho, P. Magdolenová, and M. Bugaj, "Unmanned aerial vehicle technology in the process of early fire localization of buildings," *Transp. Res. Procedia*, vol. 40, pp. 461–468, Jan. 2019.

[22] A. Imdoukh, A. Shaker, A. Al-Toukhy, D. Kablaoui, and M. El-Abd, "Semi-autonomous indoor firefighting UAV," in *Proc. 18th Int. Conf. Adv. Robot. (ICAR)*, Jul. 2017, pp. 310–315.

[23] M. Popp, S. Prophet, G. Scholz, and G. F. Trommer, "A novel guidance and navigation system for MAVs capable of autonomous collision-free entering of buildings," *Gyroscopy Navigat.*, vol. 6, no. 3, pp. 157–165, Jul. 2015.

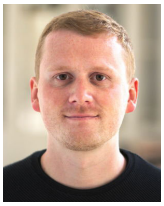
[24] M. Popp, G. Scholz, S. Prophet, and G. F. Trommer, "A laser and image based navigation and guidance system for autonomous outdoor-indoor transition flights of MAVs," in *Proc. DGON Inertial Sensors Syst. Symp. (ISS)*, Sep. 2015, pp. 1–18.

[25] S. Zhou, G. Flores, E. Bazan, R. Lozano, and A. Rodriguez, "Real-time object detection and pose estimation using stereo vision. An application for a quadrotor MAV," in *Proc. Workshop Res., Edu. Develop. Unmanned Aerial Syst. (RED-UAS)*, Nov. 2015, pp. 72–77.

[26] G. Flores, S. Zhou, R. Lozano, and P. Castillo, "A vision and GPS-based real-time trajectory planning for MAV in unknown urban environments," in *Proc. Int. Conf. Unmanned Aircr. Syst. (ICUAS)*, May 2013, pp. 1150–1155.

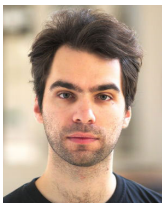
[17] https://www.mbzirc.com/winning-teams/2020

[27] R. C. Julian, C. J. Rose, H. Hu, and R. S. Fearing, "Cooperative control and modeling for narrow passage traversal with an ornithopter MAV and lightweight ground station," in *Proc. Citeseer AAMAS*, 2013, pp. 103–110.
[28] G. Loianno, C. Brunner, G. McGrath, and V. Kumar, "Estimation, control, and planning for aggressive flight with a small quadrotor with a single camera and IMU," *IEEE Robot. Autom. Lett.*, vol. 2, no. 2, pp. 404–411, Apr. 2017.
[29] D. Falanga, E. Mueggler, M. Faessler, and D. Scaramuzza, "Aggressive quadrotor flight through narrow gaps with onboard sensing and computing using active vision," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2017, pp. 5774–5781.
[30] T. Tomic, K. Schmid, P. Lutz, A. Domel, M. Kassecker, E. Mair, I. Grixa, F. Ruess, M. Suppa, and D. Burschka, "Toward a fully autonomous UAV: Research platform for indoor and outdoor urban search and rescue," *IEEE Robot. Autom. Mag.*, vol. 19, no. 3, pp. 46–56, Sep. 2012.
[31] K. Schmid, P. Lutz, T. Tomić, E. Mair, and H. Hirschmüller, "Autonomous vision-based micro air vehicle for indoor and outdoor navigation," *J. Field Robot.*, vol. 31, no. 4, pp. 537–570, Jul. 2014.
[32] T. Baca, M. Petrlik, M. Vrba, V. Spurny, R. Penicka, D. Hert, and M. Saska, "The MRS UAV system: Pushing the frontiers of reproducible research, real-world deployment, and education with autonomous unmanned aerial vehicles," 2020, *arXiv:2008.08050*. [Online]. Available: http://arxiv.org/abs/2008.08050
[33] T. Baca, D. Hert, G. Loianno, M. Saska, and V. Kumar, "Model predictive trajectory tracking and collision avoidance for reliable outdoor deployment of unmanned aerial vehicles," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Oct. 2018, pp. 1–8.
[34] T. Lee, M. Leok, and N. H. McClamroch, "Geometric tracking control of a quadrotor UAV on SE(3)," in *Proc. 49th IEEE Conf. Decis. Control (CDC)*, Dec. 2010, pp. 5420–5425.
[35] S. Kohlbrecher, O. von Stryk, J. Meyer, and U. Klingauf, "A flexible and scalable SLAM system with full 3D motion estimation," in *Proc. IEEE Int. Symp. Saf., Secur., Rescue Robot.*, Nov. 2011, pp. 155–160.
[36] A. Siadat, A. Kaske, S. Klausmann, M. Dufaut, and R. Husson, "An optimized segmentation method for a 2D laser-scanner applied to mobile robot navigation," *IFAC Proc. Volumes*, vol. 30, no. 7, pp. 149–154, Jun. 1997.
[37] G. Welch and G. Bishop, "An introduction to the Kalman filter," in *Proc. SIGGRAPH*, 2006, p. 16.
[38] V. Usenko, L. von Stumberg, A. Pangercic, and D. Cremers, "Real-time trajectory replanning for MAVs using uniform B-splines and a 3D circular buffer," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Sep. 2017, pp. 215–222.
[39] W. Minkina and S. Dudzik, *Infrared Thermography: Errors and Uncertainties*. Hoboken, NJ, USA: Wiley, 2009.
[40] P. Schillinger, S. Kohlbrecher, and O. von Stryk, "Human-robot collaborative high-level control with application to rescue robotics," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2016, pp. 2796–2802.
[41] M. Quigley, K. Conley, B. Gerkey, J. Faust, T. Foote, J. Leibs, R. Wheeler, and A. Y. Ng, "ROS: An open-source robot operating system," in *Proc. Workshop Open Source Softw. (ICRA)*, vol. 3, 2009, p. 5.



VOJTECH SPURNY received the M.Sc. degree from Czech Technical University in Prague, Czech Republic, where he is currently pursuing the Ph.D. degree in methods of planning and coordination for unmanned aerial vehicles. Since 2014, he has been a part of the Multi-Robot Systems Laboratory with CTU Prague. He was a Key Member of CTU-UPenn-UoL and CTU-UPENN-NYU teams in the MBZIRC 2017 and 2020 robotic competitions in Abu Dhabi, United Arab Emirates.

VACLAV PRITZL received the M.Sc. degree from Czech Technical University in Prague, Czech Republic, where he is currently pursuing the Ph.D. degree focused on data fusion in aerial robotics. He focuses on methods enabling indoor exploration for unmanned aerial vehicles. Since 2017, he has been a part of the Multi-Robot Systems Laboratory with CTU Prague. He was a member of the CTU-UPENN-NYU team in the MBZIRC 2020 competition.



VIKTOR WALTER received the M.Sc. degree from the Brno University of Technology, Czech Republic, where he is currently pursuing the Ph.D. degree in mutual relative localization of unmanned aerial vehicles. Since 2016, he has been a part of the Multi-Robot Systems Laboratory with CTU Prague. He is primarily focused on onboard computer vision for unmanned aerial vehicles and behaviors based on such inputs. He was also a member of the CTU-UPENN-NYU team in the MBZIRC 2020 competition.

MATEJ PETRLIK received the M.Sc. degree from Czech Technical University in Prague, Czech Republic, where he is currently pursuing the Ph.D. in sensor fusion, cooperative mapping, and dynamic state estimation of unmanned aerial vehicles. He was also a member of CTU-UPenn-UoL and CTU-UPENN-NYU teams in the MBZIRC 2017 and 2020, and the CTU-CRAS-NORLAB team in the DARPA SubT competition.

TOMAS BACA (Graduate Student Member, IEEE) received the M.Sc. degree from Czech Technical University in Prague, Czech Republic, where he is currently pursuing the Ph.D. degree in distributed remote sensing. Since 2014, he has been a part of the Multi-Robot Systems Laboratory with CTU Prague. He focuses on distributed radiation sensing with unmanned aerial vehicles and planning and control for unmanned aerial vehicles. He was a Key Member of CTU-UPenn-UoL and CTU-UPENN-NYU teams in the MBZIRC 2017 and 2020 robotic competitions in Abu Dhabi.

PETR STEPAN received the M.Sc. degree from Charles University, Czech Republic, in 1994, and the Ph.D. degree from Czech Technical University in Prague, Czech Republic. Since 1996, he has been a Research Fellow with Czech Technical University in Prague. He focuses on sensor fusion, mapping, and localization tasks. He participates in six EU project about autonomous robot. He was also a member of CTU-UPenn-UoL and CTU-UPENN-NYU teams in the MBZIRC 2017 and 2020.

DAVID ZAITLIK received the M.Sc. degree from Czech Technical University in Prague, Czech Republic. He is currently a Research Fellow focused on embedded systems. Since 2019, he has been a part of the Multi-Robot Systems Laboratory with CTU Prague. He was a member of the CTU-UPENN-NYU team in the MBZIRC 2020 competition.

MARTIN SASKA (Member, IEEE) received the M.Sc. degree from Czech Technical University in Prague, in 2005, and the Ph.D. degree from the University of Wuerzburg, Germany. Since 2009, he has been a Research Fellow with Czech Technical University in Prague, where he has founded and since leads the Multi-Robot Systems Laboratory and co-founded the Center for Robotics and Autonomous Systems. He was a Visiting Scholar with the University of Illinois at Urbana-Champaign, Champaign, IL, USA, in 2008, and the University of Pennsylvania, Philadelphia, PA, USA, in 2012, 2014, and 2016. He is the author or a coauthor of more than 70 publications in peer-reviewed conferences and more than 20 publications in impacted journals.

Chapter 4

Localization, Grasping, and Transportation of Magnetic Objects by a team of MAVs in Challenging Desert-like Environments

This section presents the second core publication called *Localization, Grasping, and Transportation of Magnetic Objects by a team of MAVs in Challenging Desert-like Environments* [2c] published in Q1 journal IEEE Robotics and Automation Letters in 2018.

[2c] G. Loianno, **V. Spurny**, J. Thomas, *et al.*, "Localization, Grasping, and Transportation of Magnetic Objects by a team of MAVs in Challenging Desert-like Environments," *IEEE Robotics and Automation Letters*, vol. 3, no. 3, pp. 1576–1583, 2018

The **introductory** section of the paper states contributions and presents related work focused on aerial manipulation and transport. To the best of our knowledge, this was the first time that estimation, planning, and control problems for multiple UAVs are addressed simultaneously in a challenging scenario, such as in a desert.

The following section **platform design** introduces selected robot hardware equipment in detail, such as base frame, motors, autopilot, onboard computer, and sensor suite (GPS, RGB camera, and range-finder). This section also includes an open-hardware gripper design for reuse by the community.

The next three sections describe **control**, **trajectory planning**, and **state estimation**. The following **experimental results** section shows data obtained from an experimental campaign in a desert in the United Arab Emirates, where a team of three UAVs were deployed. These results and observations are summarized in the last section of the paper.

The contribution to this manuscript by the author of this thesis is equivalent to the main author. The author of this thesis contributed primarily by designing a strategy for cooperatively solving the task with a focus on problematic communication infrastructure in the desert. Furthermore, the author helped with designing and implementing key components of the proposed solution, such as trajectory planning and communication.

CHAPTER 4. LOCALIZATION, GRASPING, AND TRANSPORTATION OF MAGNETIC OBJECTS BY A TEAM OF MAVs IN CHALLENGING DESERT-LIKE ENVIRONMENTS

This is the author's version of an article that has been published. Changes were made to this version by the publisher prior to publication.
The final version of paper is available at https://doi.org/10.1109/LRA.2018.2800121

IEEE ROBOTICS AND AUTOMATION LETTERS. PREPRINT VERSION. ACCEPTED JANUARY, 2018

Localization, Grasping, and Transportation of Magnetic Objects by a team of MAVs in Challenging Desert like Environments

Giuseppe Loianno* [1], Vojtech Spurny* [2], Justin Thomas* [1], Tomas Baca* [2], Dinesh Thakur* [1], Daniel Hert* [1], Robert Penicka* [2], Tomas Krajnik[3], Alex Zhou[1], Adam Cho[1], Martin Saska[2], and Vijay Kumar[1]

***Abstract*—Autonomous Micro Aerial Vehicles have the potential to assist in real life tasks involving grasping and transportation, but not before solving several difficult research challenges. In this work, we address the design, control, estimation, and planning problems for cooperative localization, grasping, and transportation of objects in challenging outdoor scenarios. We demonstrate an autonomous team of MAVs able to plan safe trajectories for manipulation of ferrous objects, while guaranteeing inter-robot collision avoidance and automatically creating a map of the objects in the environment. Our solution is predominantly distributed, allowing the team to pick and transport ferrous disks to a final destination without collisions. This result is achieved using a new magnetic gripper with a novel feedback approach, enabling the detection of successful grasping. The gripper design and all the components to build a platform are clearly provided as open-source hardware for reuse by the community. Finally, the proposed solution is validated through experimental results where difficulties include inconsistent wind, uneven terrain, and sandy conditions.**

***Keywords*—Aerial Systems: Applications, Field Robots, Swarms**

I. Introduction

Micro Aerial Vehicles (MAVs) equipped with onboard sensors are ideal platforms for autonomous navigation in complex and confined environments for solving tasks such as exploration [1], inspection [2], mapping [3], interaction with the environment [4], [5], and search and rescue [6]. In the recent years, MAVs have been endowed with manipulation and transportation capabilities [7], [8], [9], [10]. This has contributed to increase their ability to solve tasks involving environment interaction and manipulation, without limiting their operations to surveillance and search and scenarios. In this context, the opportunity to use multiple MAVs can contribute to increase the efficiency of missions, solving multiple tasks in a cooperative manner.

Fig. 1: Navigation of three MAVs in the Abu Dhabi desert.

In this work, we address the design, control, planning, and estimation problems to cooperatively localize, acquire, and transport ferrous objects in a challenging outdoor desert scenario with uneven terrain and inconsistent winds as shown in Fig. 1. Teams of MAVs can provide additional benefits compared a single vehicle. For example, it is possible to eliminate single points of failure while achieving exploration and mapping more quickly. Although the complexity of such systems increases with the number of MAVs, the exchange of information between vehicles can be exploited by each agent in order to make better decisions, plan motions, and allocate tasks. Relevant to this work are results focusing on areas involving multi-MAVs ranging from autonomous navigation to grasping for environment interaction. Different works have analyzed the problem of aerial manipulation and transportation in indoor scenarios using cables [8], [11], [12], using rigid attachements [7], and multi-arm manipulator systems [9], [13]. All of these works rely on motion capture systems for state estimation and focus on ensuring a static equilibrium of the payload at a desired pose while respecting constraints on the tension. Other works explore similar tasks leveraging multi-MAVs for transportation of objects using cables, but with larger vehicles and in real outdoor scenarios [14], [15]. In the proposed work, grasping does not rely on cable connections but rather a rigid connection between the MAV and the object. Moreover, the systems were already deployed with connections between the agents. Conversely, our system must

Manuscript received: September, 10, 2017; Revised November, 15, 2017 Accepted January, 13, 2018.

This paper was recommended for publication by Editor J. Roberts upon evaluation of the Associate Editor and Reviewers' comments. This work was supported by the Mohamed Bin Zayed International Robotics Challenge (MBZIRC), ARL grant W911NF-08-2-0004, ONR grants N00014-07-1-0829, N00014-14-1-0510, ARO grant W911NF-13-1-0350, NSF grants IIS-1426840, IIS-1138847, DARPA grants HR001151626, HR0011516850, CTU grant SGS17/187/OHK3/3T/13, CSF project 17-16900Y, CESNET project, and CSF project 17-27006Y.

* These author contributed equally

[1]The authors are with the GRASP Lab, University of Pennsylvania, 3330 Walnut Street, Philadelphia, PA 19104, USA. email: {loiannog, jut, tdinesh, alexzhou, acho, kumar}@seas.upenn.edu.

[2]The authors are with the Department of Cybernetics, Faculty of Electrical Engineering, Czech Technical University in Prague. email: {spurny.vojtech, tomas.baca, penicrob, hertdani, saskal}@fel.cvut.cz.

[3]The author is with the Artificial Intelligence Center, Faculty of Electrical Engineering, Czech Technical University in Prague. email: {tomas.krajnik}@fel.cvut.cz.

Digital Object Identifier (DOI): see top of this page.

This is the author's version of an article that has been published. Changes were made to this version by the publisher prior to publication.
The final version of paper is available at https://doi.org/10.1109/LRA.2018.2800121

handle the additional difficulty of switching between different states ranging from navigation to finding and grasping objects in real-time.

Multi-vehicle coordination and planning problems are analyzed in [16], [17], [18], where the grasping and transportation problems are not considered. Other approaches analyze the scheduling and dynamic reallocation problems [19], [20]. In all these cases, motions were limited to indoor environments, and the works do not address the object identification and grasping tasks.

This work presents multiple contributions. First, we present how to obtain a reliable state estimation and control for each vehicle, enabling successful navigation and grasping of ferrous objects. Second, we convey the design and coordination to enable a team of aerial robots to identify and collect ferrous objects with the overarching objective to transport them to a final common destination. We also provide an open-hardware gripper design for reuse by the community. The approach introduces a compliance in the grasping mechanism, obtained using an elastic element in the join. This passive compliance is very convenient in grasping and interaction tasks with aerial vehicles as already demonstrated in multiple works [21], [22], [23]. The proposed system, including gripper and all sensors, is very affordable since it costs less than $3500 USD. Finally, this is the first time that estimation, planning, and control problems are addressed simultaneously in a challenging scenario such as in a desert. Uneven surfaces, inconsistent light conditions, and strong winds make the grasping and navigation problems even more challenging. We provide our insights and solutions to these problems, and we believe that the proposed solution, both in terms of hardware and software, can be useful for search and rescue and delivery scenarios in extreme conditions. In addition, the approach can be adopted by other groups in the community, since most of the hardware components are off-the shelf. A simplified version of the proposed approach (applied in a flat, structured environment, in comparison with the varied terrain of sand dunes presented in this work) exhibited the best performance among 20 participants in the "Treasure Hunt" challenge of the Mohamed Bin Zayed International Robotics Challenge (MBZIRC[1] 2017). This competition attracted wide interest in the robotic community (143 teams registered), was one of the motivations of this research, and served as a valuable and trustworthy benchmark for evaluation of our system's performance compared to state-of-the-art solutions.

The paper is organized as follows. Section II shows the platform design and the grasping mechanism. Section III presents our methodology to generate trajectories for multi-agent navigation while avoiding inter-robot collisions when communication is available, and Section IV discusses the dynamics and control of the vehicle. Section V explains our strategy to obtain the pose of the vehicle at a high rate for autonomous navigation, and it describes the target detection. Section VI presents our experimental results, and Section VII concludes the work and presents future opportunities.

[1]http://mbzirc.com

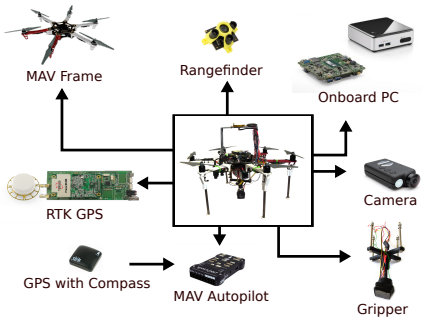


Fig. 2: The proposed system architecture.

II. Platform Design

The proposed system is depicted in Fig. 2. Each individual platform is composed of a DJI hexacopter frame F550[2], DJI E310 motors[3], a gripper, computational resources, and sensor modules. An Intel-NUC-i7[4] is used for on-board computation, a Mobius ActionCam[5] camera is employed for object detection and relative visual localization, and a TeraRanger rangefinder[6] is used for altitude estimation. The system is controlled on the lowest level by a PixHawk autopilot, which contains a set of sensors such as accelerometers, gyroscopes, and magnetometers that are necessary to stabilize the attitude of the MAV. We use a combination of GNSS (Global Navigation Satellite System) attached to the Pixhawk autopilot and Real Time Kinematic (RTK) satellite for localization. The RTK system improves the single GNSS localization module with data derived from satellite-based positioning systems such as GPS, GLONASS, Galileo, and BeiDou. Information on the position is provided in the RTK system by a PRECIS-BX305 GNSS RTK BOARD (GPS L1L2/GLONASS G1/BEIDOU B1B3), providing a final accuracy of 10 millimeters (mm) ± 1 parts-per-million (ppm) horizontally and 15 mm ± 1 ppm vertically when the RTK device is in the most accurate state, "RTK FIX". In principle, the vertical position (altitude) provided by the RTK GPS is measured above the mean sea level, which means that the MAV does not have precise information about the ground level profile and the height of objects for grasping, which is supplemented by the rangefinder measurements.

A key component in our design is the gripper, which must be lightweight and able to acquire and carry flat ferrous objects as required by the MBZIRC subchallenge. The gripper consists of 3D printed parts and includes a mounting bracket, a compression shaft, and an end effector as shown in Fig 3. The mounting bracket connects to the ventral part of the vehicle and provides a receptacle for the compression shaft.

[2]https://www.dji.com/flame-wheel-arf/feature
[3]https://www.dji.com/e310
[4]https://www.intel.com/content/www/us/en/products/boards-kits/nuc/kits/nuc7i7bnh-16gb-optane.html
[5]https://www.mobius-actioncam.com/
[6]http://www.teraranger.com/products/teraranger-one/

This is the author's version of an article that has been published. Changes were made to this version by the publisher prior to publication.
The final version of paper is available at https://doi.org/10.1109/LRA.2018.2800121

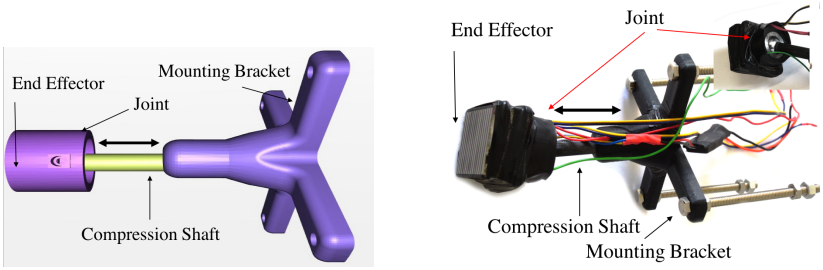


Fig. 3: The designed gripper (left) and its realization (right).

Internally, a spring allows the shaft to retract during take off and landing to avoid damaging the gripper, and it also provides axial compliance when grasping. The shaft is connected to the end effector of the gripper via a ball-and-socket joint. The end effector is the core of the gripper; an electro-permanent magnet OpenGrab EPM v3[7] that combines the advantages of electro and permanent magnets and creates a strong magnetic contact with the ferrous object. The payload capacity of the electromagnet is determined by

$$F = m_o g s \tag{1}$$

where F is the required nominal force, m_o the object mass and s is a safety factor, which needs to be properly selected still allow an effective object release, independently of the grasping location on the object surface.

We also must account for the decrease in the available holding force of the magnet, which is perpendicular to the end effector magnet plane, in case its direction is not parallel to the gravity one. It was experimentally verified that a holding force of the ferrite magnet decreases by nearly 85% in shear. Using a maximum rotation of $45°$ of the magnet results in a decrease of approximately 42% in holding strength. This is then accounted for in the above equation with the safety factor. The total weight of the proposed gripper design is 175 g. The EPM Gripper works by taking a 5 V 1 A power to switch states, and it does not require constant power. Finally, another key component is a spherical joint between the shaft and the EPM, which allows mechanical decoupling of the relative rotation between the platform and the object.

Out of the box, the EPM does not provide any feedback information required to identify a successful grasping. However, this kind of feedback is critical for good performance of the system. Without it, in the case of unsuccessful grasping, the vehicles would use precious battery time flying to the drop location and back, without carrying anything. To address this problem, two Hall effect sensors are installed on the sides of the EPM. When a ferrous object is attached, the magnetic field around the magnet changes. This is reflected by a change in the output voltage of the Hall effect sensors. To prevent false-positive detections, a short calibration is done before each grasping attempt during which 100 samples of the output voltages are averaged and experimentally a new threshold is set above the average. If the measured voltage exceeds the set threshold, a successful grasp is signaled. Likewise, if the object is dropped during flight, the measured voltage drops below the threshold and an object loss is signaled, which allows the MAV to abort the delivery and attempt grasping another object.

A custom board was designed to provide a low-level interface between the main computer, the gripper, and a serial radio module. This board uses an 8-bit ATxmega 128A4U[8] microcontroller and an FT232RL[9] USB serial driver to communicate with the main computer. The gripper is controlled by a PWM signal, and the feedback voltage from the Hall effect sensors is measured by the microcontroller's ADCs. The board is also fitted with a socket to accommodate an XBee Pro radio module, which is used to relay RTK corrections from the base-station to the RTK GPS module. A low-level architecture is shown in Fig. 4. We release the reference files to make and

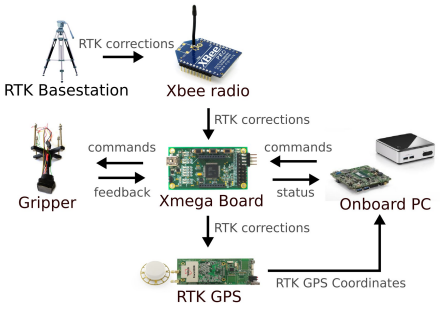


Fig. 4: The low-level interface with communication and feedback.

3D print the gripper as well as a self contained code so that anyone can interface the gripper with a μController[10].

III. Modeling and Control

The position controller uses the estimated state as feedback to follow trajectories given as an output of the high-

[7] http://nicadrone.com/index.php?id_product=72&controller=product

[8] http://www.microchip.com/wwwproducts/en/ATxmega128A4U

[9] http://www.ftdichip.com/Products/ICs/FT232R.htm

[10] https://github.com/loiannog/MBZIRCgripper

This is the author's version of an article that has been published. Changes were made to this version by the publisher prior to publication.
The final version of paper is available at https://doi.org/10.1109/LRA.2018.2800121

level trajectory planner. In many previous works, because the attitude dynamics can be assumed to be faster than the dynamics governing the position, linearized controllers are used for both loops [24], [25]. However, we need the system to be capable of large deviations from the hover configuration during operations like fast mapping of objects or for heavy wind compensation, so we use a nonlinear controller. Let us consider an inertial reference frame denoted by $[\mathbf{e}_1\ ,\ \mathbf{e}_2\ ,\ \mathbf{e}_3]$ and a body reference frame centered in the center of mass of the vehicle with an orientation denoted by $\mathbf{R} = [\mathbf{b}_1\ ,\ \mathbf{b}_2\ ,\ \mathbf{b}_3]$ where $\mathbf{R} \in SO(3)$. The dynamic model of the vehicle can be expressed as

$$\begin{aligned} \dot{\mathbf{x}} &= \mathbf{v},\ m\dot{\mathbf{v}} = f\mathbf{R}\mathbf{e}_3 - mg\mathbf{e}_3, \\ \dot{\mathbf{R}} &= \mathbf{R}\hat{\mathbf{\Omega}},\ \mathbf{J}\dot{\mathbf{\Omega}} + \mathbf{\Omega}\times\mathbf{J}\mathbf{\Omega} = \mathbf{M}, \end{aligned} \tag{2}$$

where $\mathbf{x} \in \mathbb{R}^3$ is the Cartesian position of the vehicle expressed in the inertial frame, $\mathbf{v} \in \mathbb{R}^3$ is the velocity of the vehicle in the inertial frame, $m \in \mathbb{R}$ is the mass, which includes the mass of the object during the grasping phase using the feedback mechanism described in the previous section, $f \in \mathbb{R}$ is the net thrust, $\mathbf{\Omega} \in \mathbb{R}^3$ is the angular velocity in the body-fixed frame, and $\mathbf{J} \in \mathbb{R}^{3\times3}$ is the inertia matrix with respect to the body frame. The hat symbol $\hat{}$ denotes the skew-symmetry operator according to $\hat{\mathbf{x}}\mathbf{y} = \mathbf{x}\times\mathbf{y}$ for all $\mathbf{x},\mathbf{y} \in \mathbb{R}^3$, g is the standard gravitational acceleration and $\mathbf{e}_3 = [0\quad 0\quad 1]^\top$. The total moment $\mathbf{M} \in \mathbb{R}^3$, with $\mathbf{M} = [M_1\quad M_2\quad M_3]^\top$, along all axes of the body-fixed frame and the thrust $f \in \mathbb{R}$ are control inputs of the plant. The dynamics of rotors and propellers are neglected and it is assumed that the force of each propeller is directly controlled. The total thrust, $f = \sum_{j=1}^{6} f_j$, acts in the direction of the z axis of the body-fixed frame, which is orthogonal to the plane defined by the centers of the six propellers. The relationship between a single motor thrust f_j, the net thrust f, and the moments $\mathbf{M}$ can be written as

$$\begin{bmatrix} f \\ M_1 \\ M_2 \\ M_3 \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 \\ sd & 1 & sd & -sd & -1 & -sd \\ -cd & 0 & cd & cd & 0 & -cd \\ -1 & 1 & -1 & 1 & -1 & 1 \end{bmatrix} \begin{bmatrix} f_1 \\ f_2 \\ f_3 \\ f_4 \\ f_5 \\ f_6 \end{bmatrix} \tag{3}$$

where $c = \cos(30°)$, $s = \sin(30°)$ and d is the distance from the center of mass to the center of each rotor in the $\mathbf{b}_1$, $\mathbf{b}_2$ plane. For non-zero values of d, eq. (3) can be inverted using the right pseudo-inverse.

For control, we build on the work in [26] and [27] with control inputs $f \in \mathbb{R}$ and $\mathbf{M} \in \mathbb{R}^3$ chosen as

$$\begin{aligned} f = -\Bigg(&-k_x\mathbf{e}_x - k_{ib}\mathbf{R}\int_0^t \mathbf{R}(\tau)^\top\mathbf{e}_x d\tau - k_{iw}\int_0^t \mathbf{e}_x d\tau \\ &- k_v\mathbf{e}_v - mg\mathbf{e}_3 + m\ddot{\mathbf{x}}_d\Bigg)\cdot\mathbf{R}\mathbf{e}_3, \end{aligned} \tag{4}$$

$$\begin{aligned} \mathbf{M} = &-k_R\mathbf{e}_R - k_\Omega\mathbf{e}_\Omega + \mathbf{\Omega}\times J\mathbf{\Omega} \\ &- J\left(\hat{\mathbf{\Omega}}R^\top R_c\mathbf{\Omega}_c - \mathbf{R}^\top\mathbf{R}_c\dot{\mathbf{\Omega}}_c\right), \end{aligned} \tag{5}$$

with $\ddot{\mathbf{x}}_d$ the desired acceleration, k_x, k_{ib}, k_v, k_R, k_Ω positive definite terms. We extend the referenced controllers by including two integral terms, which accumulate error in the body frame and the world frame, respectively. In another work [28], an integral action is considered in the world frame. These terms are extremely important considering the operating conditions of strong and inconsistent wind, for example around dunes in the desert. Both terms provide the opportunity to capture external disturbances (*e.g.,* wind) separately from internal disturbances (*e.g.,* an inefficient prop or a payload imbalance), particularly when the vehicle is permitted to yaw or rotate about the vertical axis. The thrust and moments are then converted to motor rates according to the characteristic of the proposed vehicle. The subscript C denotes a commanded value and $\mathbf{R}_C = [\mathbf{b}_{1,C},\quad \mathbf{b}_{2,C},\quad \mathbf{b}_{3,C}]$ is calculated as

$$\begin{aligned} \mathbf{b}_{2,des} &= [-\sin\psi_{des},\quad \cos\psi_{des},\quad 0]^\top,\ \mathbf{b}_{3,C} = \frac{\mathbf{f}}{||\mathbf{f}||}, \\ \mathbf{b}_{1,C} &= \frac{\mathbf{b}_{2,des}\times\mathbf{b}_3}{||\mathbf{b}_{2,des}\times\mathbf{b}_3||},\ \mathbf{b}_{2,C} = \mathbf{b}_3\times\mathbf{b}_1, \\ \hat{\mathbf{\Omega}}_C &= \mathbf{R}_C^\top\dot{\mathbf{R}}_C. \end{aligned} \tag{6}$$

where $\dot{\mathbf{R}}_C$. is obtained differentiating with respect to time the vector elements composing $\mathbf{R}_C$. Note that here we have to define $\mathbf{b}_{2,des}$ based on the yaw instead of defining $\mathbf{b}_{1,des}$ as done in [27] due to a different Euler angle convention (we use the ZYX convention instead of ZXY). The error quantities $\mathbf{e}_R$, $\mathbf{e}_\Omega$, $\mathbf{e}_x$, $\mathbf{e}_v$ are defined in [26], and they represent the orientation, angular rate errors, and translation errors respectively. The symbol $.^\vee$ represent the *vee* map $\mathfrak{so}(3) \to \mathbb{R}^3$. The exponential stability has been proved for a similar PID controller in [28].

IV. Trajectory Planning

Given a pre-defined rectangular area to scan by a team of MAVs, the task first involves generating collision free trajectories for each MAV. Secondly, we must avoid inter-MAV collisions when grasped objects have to be dropped in a common assigned box. A common way for area coverage is to use a Zamboni pattern or a lawn mover pattern. Path planning for such a problem is described as Coverage Path Planning (CPP) [29], where for a given area the CPP provides a path from which the entire workplace can be scanned with an onboard camera. The work in [30] combines graph search techniques and spline-based methods to provide a solution for multi-robot coverage with requirements for sensor resolution and field-of-view.

Instead of solving a combined multi-robot path planning coverage problem, we choose a simple area decomposition method and then assign an independent zone for each MAV to localize and collect objects. The required coverage area is split into equally-sized zones and guarantees that no paths intersect during the coverage phase. Each MAV plans the coverage path using the Boustrophedon coverage [31] in its zone as shown in Fig. 5. A reduced field of view is used for the planning and is set based on the required overlap in the coverage (set

This is the author's version of an article that has been published. Changes were made to this version by the publisher prior to publication.
The final version of paper is available at https://doi.org/10.1109/LRA.2018.2800121

to 20%) and on the real FOV camera projection to the ground plane with respect to the sweeping altitude.

While following the trajectory, the MAV detects and localizes the position of any object. Upon a detection, the trajectory tracking is stopped and the MAV tries to grasp the object immediately. After either a successful grasping and dropping of the object or after a number of unsuccessful grasp attempts, the MAV continues sweeping from the last point on the trajectory.

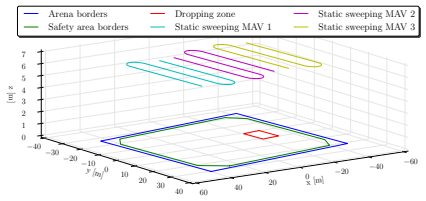


Fig. 5: Coverage trajectories based on Boustrophedon coverage with a Dubins vehicle [32] and area decomposition into three distinct zones, one for each MAV.

The coverage trajectories are collision free since the robots are in independent zones. However, a MAV may need to pass through zones belonging to other robots in order to drop object after its successful grasping and as well in order to return back into its assigned zone. During this maneuvers, we take a proactive approach to ensure that a collision will not occur. There are many challenges to develop such an onboard collision free planner. Our planner uses a simple idea with different altitudes for each MAV. However, while completing the given task, the MAVs cannot stay only in these altitudes during the mission, as they have to descend for events such as grasping and consequent dropping of the objects. The position of the MAV in x-axis and y-axis does not alter rapidly in these events because they are realized by following strictly vertical trajectories. Therefore, the MAV after leaving its assigned altitude is assumed to be a static obstacle for other MAVs and the finding the shortest free path between the starting and goal points may be easily realized by a visibility graph method [33].

This strategy works due to sharing the estimated positions of MAVs between robots. In the case of a communication dropout, typically occurring during deployment of the multi robot system in real world conditions, using a different altitude for each MAV helps to minimize the likelihood of a collision. Furthermore, since only one dropping zone is available, the MAVs must coordinate themselves so that the dropping zone is only occupied by one MAV at a time. When there is a communication dropout, a simple time-window based approach is used to resolves this issue.

V. State Estimation

The state estimation algorithm must identify the state of each robot and the location of the objects to be grasped.

A. State Estimation for Control

As shown in Section II, the platform is equipped with several different sensors such as GPS, a height sensor, and cameras. These sensors can be fused using the complementary nature of each sensor in terms of accuracy and speed to obtain a reliable state estimate to control the MAV. The Extended Kalman Filter available on the PixHawk fuses the inertial sensors, altitude pressure sensor, and the GPS receiver. However, for the considered task, the precision and accuracy for grasping is extremely important since the main goal is to be able to transport the objects to a final destination. The position estimate in the lateral axes is based on the estimate provided by PixHawk, namely positions $\mathbf{x}^p$, and velocities $\dot{\mathbf{x}}^p$. Although, its precision might be satisfactory locally, it is prone to heavy drift in short time ranges. This drift is corrected by differential RTK GPS to ensure repeatability of experiments. Position measurements from the RTK GPS receiver are fused using a Linear Kalman Filter with model

$$\mathbf{A} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}, \mathbf{B} = \begin{pmatrix} \Delta t \\ \Delta t \end{pmatrix}, \tag{7}$$

where $\mathbf{x}^e_{[n]} = \mathbf{A}\mathbf{x}^e_{[n-1]} + \mathbf{B}\mathbf{u}_{[n-1]}$ is the linear system equation, $\mathbf{x}^e_{[n]} = (x, y)^\top_{[n]}$ is the state vector finally used for control, and $\mathbf{u}_{[n]}$ is the system input. The input vector $\mathbf{u}$ consists of velocities obtained by differentiating positions $\mathbf{x}^p$, which ensures that our filter does not introduce any more drift to the resulting estimate when no RTK GPS corrections are involved due to inaccuracies of $\dot{\mathbf{x}}^p$. In situations when the position is not being corrected, the resulting estimate follows the same relative state trajectory as $\mathbf{x}^p$, just shifted according to the last correction.

The multi-robot scenario requires a common reference frame between all vehicles. The base of our Cartesian system is set to predefined GPS coordinates, and its orientation is according to the *ENU* convention, thus the 1st, 2nd and 3rd axis points to the East, North and Upwards respectively. This is the convention used in Section III to define the inertial frame. A point of origin is measured using the RTK GPS, to which all independent coordinate systems of all MAVs are then shifted after each of them is powered up. The common base station of the differential RTK GPS then ensures that all MAVs estimates are corrected to lie within the same global coordinates. In this way, the estimation of the altitude does not only rely on the pressure sensor, which is not accurate. A Linear Kalman Filter is used to incorporate altitude corrections from the differential RTK GPS, the downward-facing TeraRanger height sensor, and the object detector, which is able to provide an estimate of the relative distance when flying above an object. The estimator provides an option to switch between those sources of data depending on the current task and the circumstances. Correcting the altitude using the TeraRanger rangefinder is feasible when flying above uneven ground, but it cannot be reliably used when the downward-facing sensor is obstructed e.g., when carrying an object, or when there might be a foreign object on the ground. On the other hand, the RTK GPS is able to provide precise relative altitude measurements, but only when RTK FIX has been established, which depends on GNSS signal strength and the quality of the communication link between the base station and the MAV. Further, it is unaware of the ground profile, so it cannot directly provide an estimate of the

This is the author's version of an article that has been published. Changes were made to this version by the publisher prior to publication.
The final version of paper is available at https://doi.org/10.1109/LRA.2018.2800121

Fig. 6: One of the vehicle during a grasping task in the desert (on the left) and two vehicles during the navigation task after grasped one object respectively (on the right), which are circled in green.

height above the ground. Finally, correcting the altitude using data from the *object detector* might bring in unexpected steps in the signal due to false positive detections. To be robust to unreliable measurements, which can compromise the mission in such critical settings, a safety mechanism for detecting anomalies is used which can toggle off any of the altitude sensors from being fused.

B. Object Detection and Localization

As mentioned in Section II, the Mobius camera is used to detect the colored, round objects. This high-resolution camera has a rolling shutter, so vibrations induced by the drone motors can cause noisy effects in the image which makes the use of geometry-based methods for object detection (e.g. Hough transform) problematic [34]. Thus, we designed a computationally efficient ellipse detection algorithm, which relies on the use of statistics that are robust to this type of noise [35]. The experiments have shown that a better object detection performance is achieved with a high-resolution rolling shutter camera than with a lower-resolution, global shutter camera of similar price.

Unlike the original method [35], which used adaptive thresholding to detect black-and-white patterns, our pixel labeling was based on a 3D RGB look-up grid, which is either preloaded or build semi-automatically during a calibration procedure. During calibration, the MAV hovers (or is held by hand) over the objects in a known spatial configuration and displays the results of the detection on a GUI. The system operator can indicate the object positions and false detections, creating and refining a Hue-Saturation-Value Gaussian mixture model of the object and background colors. Once the user indicates that the detection performance is satisfactory, each cell of the RGB look-up grid (corresponding to the given color) is classified as an object or background using the Gaussian mixture model, and the look-up grid is saved. The main advantage of the RGB look-up grid is that it can implement any pixel-wise classification method in a very efficient way – classification of a given pixel is performed simply by retrieving its class from the cell that corresponds to the pixel's RGB values.

The detection algorithm is based on [35], which searches for continuous segments of object-colored pixels, establishes their bounding box, number of pixels, centroid, convexity, and compactness, and uses these statistics to reject segments that cannot correspond to circular objects. Finally, using the known object size, elliptical shape and camera parameters, the method calculates the relative 3D position of the object which is transformed into the inertial coordinate frame. If the detected object's position and the ground distance obtained from the robot's state estimate are not within a threshold, then the object is marked as a false positive. For details on the method, please refer to [36]. Targets that are trying to be grasped are filtered out in other MAVs to prevent simultaneous grasping of the same target. The following empirical laws are adopted during the manipulation task

- Objects which have not been seen for more than 5 seconds are *deactivated*.
- Objects which are deactivated for more than 3 seconds are deleted from the map.
- Measurements from the *object detector* are paired with objects in the map using a min-distance bipartite graph matching, constrained by the color of the objects.
- Objects located outside of the working area are deleted from the map, and new measurements in such areas are discarded.

Additionally, the grasping attempts might not be successful at all times. The approach also allows a temporary ban for 30 s in a 4 m radius around a particular object to avoid deadlock in the grasping state machine.

VI. Experimental Results

In this section, we present results obtained in the demanding environment of the desert of Abu Dhabi in the United Arab Emirates as shown in Fig. 6. The key objective of this work is to have a team of MAVs identify, grasp, and deliever a number of objects in an environment with an unknown ground profile, difficult wind, and unreliable communication.

The entire pipeline has been implemented in ROS, which is also responsible for the management of communication between MAVs. ROS by design is not suitable for distributed implementation, so we make use of the ROS multimaster package [37] that allows managing nodes, topics, services, parameters across different robots. To reduce load in the communication channels, only selected information (topics) are shared with other team members such as the MAV position in the global coordinate system, the current state of the high-level

This is the author's version of an article that has been published. Changes were made to this version by the publisher prior to publication.
The final version of paper is available at https://doi.org/10.1109/LRA.2018.2800121

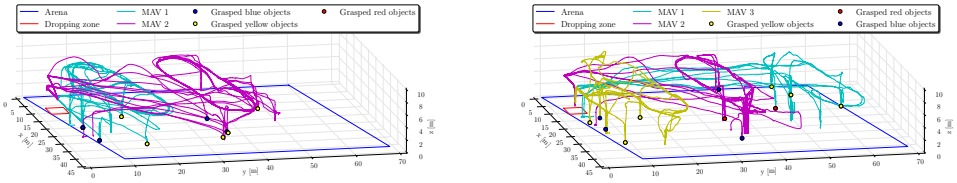
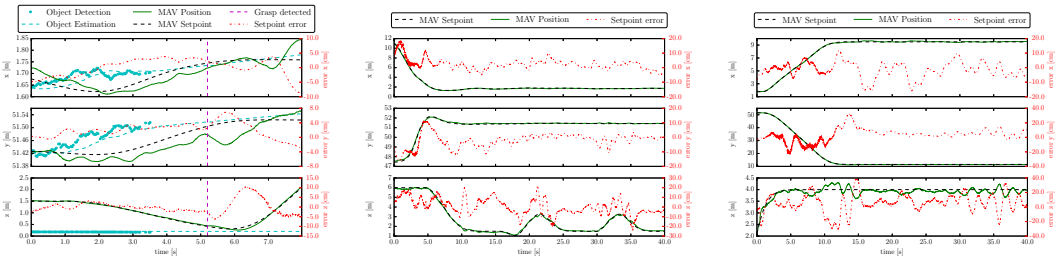


Fig. 7: Path for different vehicles during the grasping task of two different (on the left) and three different types of objects (on the right).



(a) During grasping attempt **(b)** Before grasping attempt **(c)** After successful grasping

Fig. 8: Grasping per cartesian axis with object estimation and MAV position, setpoint and setpoint errors.

state machine, the position of the object during grasping, and the planned trajectory. This information is used in nodes for collision free planning, fail-safe reactive collision avoidance, and object estimation.

The proposed approach has been tested with a set of 2 and 3 MAVs as shown in Fig. 7 on the left and right, respectively. The weight of each object is 70 g and is placed close to the ground at different altitudes (see Fig. 6 right indicated with green ellipses) in a total area of 70×45 m^2. Each robot travels an average of 250 m in the x and y Cartesian axes at an average height of 10 m. In Fig. 7, each object is identified by the color corresponding to the vehicle that performed the grasping. We can clearly identify a total of 12 objects. Each vehicle picked up and delivered at least 3 objects thus showing robustness of the system across platforms. The dropping area is identified in the top left corner of Fig. 7 by a red square on the ground. During the experiments, the robots experienced wind speeds up to 10 m/s. Regardless, over the course of 10 trials, the system achieved an average success rate of 90% when grasping and 95% when dropping objects in the goal location.

We finally discuss the control performance during the navigation and grasping task. In Fig. 8, we report on the results for the grasping task executed by one of the platform. The vehicle starts at 1.5 m above the object. During the descent phase, which starts at 1 s, the vehicle tries to center with respect to the object on the other two Cartesian axes. The control errors are, on all axes, within few centimetres, which is sufficient to successfully grasp the objects. The attached multmedia material shows many trials with similar results independently of the color and position on the terrain of the objects. In addition, in Table I, we report the average mean tracking errors before and after the grasping task. The errors are equivalent on the same magnitude range, showing that our control strategy is robust with respect to the weight changes introduced by the payload.

Average Error	Before Grasping (m)	After Grasping (m)
x	0.032	0.045
y	0.038	0.071
z	0.067	0.110

TABLE I: Average position mean errors in meters during navigation before and after the grasping task.

VII. Conclusion

In this work, we developed an approach to enable a team of MAVs to identify, grasp, and deliver objects in challenging desert-like environments. The key challenges included creating a robust gripper for ferrous objects, scanning and locating the objects of interest, uneven terrain, inconsistent wind, and the potential for inter-robot collisions, all of which make the proposed task extremely difficult. We showed the effectiveness of our approach in real tests with the ability to have multiple vehicles simultaneously collecting different colored ferrous objects in the desert. We release the main platform hardware components and designs as open source, which is useful for other researchers to create an autonomous flying robots.We believe that the proposed solution, both in terms of hardware and algorithms, will be useful in industrial automation and search and rescue scenarios. Future strategies will investigate how to carry larger magnetic objects with multiple vehicles in a cooperative manner. We will also explore improved state estimation approaches, eliminating the need for GPS, which would open up opportunities for GPS-denied tasks.

This is the author's version of an article that has been published. Changes were made to this version by the publisher prior to publication.
The final version of paper is available at https://doi.org/10.1109/LRA.2018.2800121

REFERENCES

[1] T. Tomic, K. Schmid, P. Lutz, A. Domel, M. Kassecker, E. Mair, I. Grixa, F. Ruess, M. Suppa, and D. Burschka, "Toward a fully autonomous uav: Research platform for indoor and outdoor urban search and rescue," *IEEE Robotics Automation Magazine*, vol. 19, no. 3, pp. 46–56, Sept 2012.

[2] T. Ozaslan, G. Loianno, J. Keller, C. J. Taylor, V. Kumar, J. M. Wozencraft, and T. Hood, "Autonomous navigation and mapping for inspection of penstocks and tunnels with mavs," *IEEE Robotics and Automation Letters*, vol. 2, no. 3, pp. 1740–1747, July 2017.

[3] G. Loianno, J. Thomas, and V. Kumar, "Cooperative localization and mapping of mavs using rgb-d sensors," in *IEEE International Conference on Robotics and Automation*, May 2015, pp. 4021–4028.

[4] F. Forte, R. Naldi, and L. Marconi, "Impedance Control of an Aerial Manipulator," in *American Control Conference (ACC)*, Montreal, Canada, 2012, pp. 3839–3844.

[5] J. Thomas, G. Loianno, K. Daniilidis, and V. Kumar, "Visual servoing of quadrotors for perching by hanging from cylindrical objects," *IEEE Robotics and Automation Letters*, vol. 1, no. 1, pp. 57–64, Jan 2016.

[6] N. Michael, S. Shen, K. Mohta, V. Kumar, K. Nagatani, Y. Okada, S. Kiribayashi, K. Otake, K. Yoshida, K. Ohno, E. Takeuchi, and S. Tadokoro, "Collaborative Mapping of an Earthquake-Damaged Building via Ground and Aerial Robots," *Journal of Field Robotics*, vol. 29, no. 5, pp. 832–841, 2012.

[7] R. Ritz and R. D'Andrea, "Carrying a flexible payload with multiple flying vehicles," in *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems*, Nov 2013, pp. 3465–3471.

[8] N. Michael, J. Fink, and V. Kumar, "Cooperative manipulation and transportation with aerial robots," *Autonomous Robots*, vol. 30, no. 1, pp. 73–86, Jan 2011.

[9] F. Ruggiero, M. A. Trujillo, R. Cano, H. Ascorbe, A. Viguria, C. Perz, V. Lippiello, A. Ollero, and B. Siciliano, "A multilayer control for multirotor uavs equipped with a servo robot arm," in *2015 IEEE International Conference on Robotics and Automation (ICRA)*, May 2015, pp. 4014–4020.

[10] G. Loianno and V. Kumar, "Cooperative transportation using small quadrotors using monocular vision and inertial sensing," *IEEE Robotics and Automation Letters*, vol. 3, no. 2, pp. 680–687, April 2018.

[11] J. Fink, N. Michael, S. Kim, and V. Kumar, "Planning and control for cooperative manipulation and transportation with aerial robots," *The International Journal of Robotics Research*, vol. 30, no. 3, pp. 324–334, 2011.

[12] Q. Lindsey, D. Mellinger, and V. Kumar, "Construction with quadrotor teams," *Autonomous Robots*, vol. 33, no. 3, pp. 323–336, Oct 2012.

[13] A. E. Jimenez-Cano, J. Martin, G. Heredia, A. Ollero, and R. Cano, "Control of an aerial robot with multi-link arm for assembly tasks," in *2013 IEEE International Conference on Robotics and Automation*, May 2013, pp. 4916–4921.

[14] M. Bernard and K. Kondak, "Generic slung load transportation system using small size helicopters," in *2009 IEEE International Conference on Robotics and Automation*, May 2009, pp. 3258–3264.

[15] M. Bernard, K. Kondak, I. Maza, and A. Ollero, "Autonomous transportation and deployment with aerial robots for search and rescue missions," *Journal of Field Robotics*, vol. 28, no. 6, pp. 914–931, 2011.

[16] M. Saska, T. Baca, J. Thomas, J. Chudoba, L. Preucil, T. Krajnik, J. Faigl, G. Loianno, and V. Kumar, "System for deployment of groups of unmanned micro aerial vehicles in gps-denied environments using onboard visual relative localization," *Autonomous Robots*, vol. 41, no. 4, pp. 919–944, Apr 2017.

[17] M. Turpin, N. Michael, and V. Kumar, "Capt: Concurrent assignment and planning of trajectories for multiple robots," *The International Journal of Robotics Research*, vol. 33, no. 1, pp. 98–112, June 2014.

[18] G. Loianno, Y. Mulgaonkar, C. Brunner, D. Ahuja, A. Ramanandan, M. Chari, S. Diaz, and V. Kumar, "A swarm of flying smartphones," in *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Oct 2016, pp. 1681–1688.

[19] N. Kamra and N. Ayanian, "Dynamic resource reallocation for robots on long term deployments," in *IEEE International Conference on Automation Science and Engineering*, Gothenburg, Sweden, Aug 2015, pp. 612 – 617.

[20] N. Ayanian, D. Rus, and V. Kumar, "Decentralized multirobot control in partially known environments with dynamic task reassignment," in *IFAC Workshop on Distributed Estimation and Control in Networked Systems*, Santa Barbara, CA, Sept. 2012, pp. 311–316.

[21] T. Bartelds, A. Capra, S. Hamaza, S. Stramigioli, and M. Fumagalli, "Compliant aerial manipulators: Toward a new generation of aerial robotic workers," *IEEE Robotics and Automation Letters*, vol. 1, no. 1, pp. 477–483, Jan 2016.

[22] A. Suarez, P. Ramon-Soria, G. Heredia, B. C. Arrue, and A. Ollero, "Anthropomorphic, compliant and lightweight dual arm system for aerial manipulation," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, September 2017.

[23] P. E. I. Pounds and A. M. Dollar, "Stability of helicopters in compliant contact under pd-pid control," *IEEE Transactions on Robotics*, vol. 30, no. 6, pp. 1472–1486, Dec 2014.

[24] N. Michael, D. Mellinger, Q. Lindsey, and V. Kumar, "The Grasp Multiple Micro–UAV Test Bed," *IEEE Robotics and Automation Magazine*, vol. 17, no. 3, pp. 56–65, 2010.

[25] S. Weiss, D. Scaramuzza, and R. Siegwart, "Monocular-SLAM-based navigation for autonomous micro helicopters in GPS denied environments," *Journal of Field Robotics*, vol. 28, no. 6, pp. 854–874, 2011.

[26] T. Lee, M. Leoky, and N. H. McClamroch, "Geometric tracking control of a quadrotor uav on se(3)," in *49th IEEE Conference on Decision and Control (CDC)*, Dec 2010, pp. 5420–5425.

[27] D. Mellinger and V. Kumar, "Minimum Snap Trajectory Generation and Control for Quadrotors," in *IEEE International Conference on Robotics and Automation*, Shangai, China, 2011, pp. 2520–2525.

[28] F. Goodarzi, D. Lee, and T. Lee, "Geometric nonlinear pid control of a quadrotor uav on se(3)," in *2013 European Control Conference (ECC)*, July 2013, pp. 3845–3850.

[29] E. Galceran and M. Carreras, "A survey on coverage path planning for robotics," *Robotics and Autonomous Systems*, vol. 61, no. 12, pp. 1258 – 1276, 2013.

[30] J. Keller, D. Thakur, M. Likhachev, J. Gallier, and V. Kumar, "Coordinated path planning for fixed-wing uas conducting persistent surveillance missions," *IEEE Transactions on Automation Science and Engineering*, vol. 14, no. 1, pp. 17–24, Jan 2017.

[31] H. Choset and P. Pignon, *Coverage Path Planning: The Boustrophedon Cellular Decomposition*. Springer London, 1998, pp. 203–209.

[32] L. E. Dubins, "On curves of minimal length with a constraint on average curvature, and with prescribed initial and terminal positions and tangents," *American Journal of Mathematics*, pp. 497–516, 1957.

[33] T. Lozano-Pérez and M. A. Wesley, "An algorithm for planning collision-free paths among polyhedral obstacles," *Commun. ACM*, vol. 22, no. 10, pp. 560–570, Oct. 1979.

[34] D. Afolabi, K. L. Man, H.-N. Liang, S.-U. Guan, and T. Krilavičius, "1543. monocular line tracking for the reduction of vibration induced during image acquisition." *Journal of Vibroengineering*, vol. 17, no. 2, 2015.

[35] T. Krajník, M. Nitsche, J. Faigl, P. Vaněk, M. Saska, L. Přeučil, T. Duckett, and M. Mejail, "A practical multirobot localization system," *Journal of Intelligent & Robotic Systems*, vol. 76, no. 3-4, pp. 539–562, 2014.

[36] P. Štěpán, T. Krajník, M. Petrlík, and M. Saska, "Vision techniques for on-board detection, following and mapping of moving targets," *Journal of Field Robotics*, 2017, (in review in the special issue on the MBZIRC contest).

[37] S. Hernández and F. Herrero, "Multi-master ros systems," Institut de Robtica i Informtica Industrial, CSIC-UPC, Tech. Rep., 2015.

Chapter 5

Cooperative Autonomous Search, Grasping, and Delivering in a Treasure Hunt Scenario by a Team of UAVs

This section presents the third core publication of this compiled Ph.D. thesis, *Cooperative Autonomous Search, Grasping, and Delivering in a Treasure Hunt Scenario by a Team of UAVs* [3c], which was published in the Q1 journal, Journal of Field Robotics in 2019. This 24-page publication extends the previous manuscript [2c] with a more detailed methodology description. Furthermore, it displays its functionality during multiple deployments, including Challenge 3 of the MBZIRC 2017 competition, where we achieved first place among the 143 teams registered for the competition.

[3c] **V. Spurny**, T. Baca, M. Saska, *et al.*, "Cooperative Autonomous Search, Grasping, and Delivering in a Treasure Hunt Scenario by a Team of UAVs," *Journal of Field Robotics*, vol. 36, no. 1, 125–148, 2019

During the time of writing the publication, only two solutions for solving the complete task of multi-UAV objects delivery were available [82], [83]. In the **introductory** section of the paper, the novelty of the proposed methodology is compared with these two most relevant papers. Mainly a novel control pipeline that can compensate for external factors (such as wind, which is a common disturbance in outdoor environments) and alternative strategies in the event of a communication blackout are highlighted in the paper. Furthermore, the contribution is compared to similar works focused on the related tasks of object detection, grasping, and transportation.

The second section of the paper details the proposed hardware setup for this task. The following **software system structure** section focuses on a detailed description of the individual software parts of the system. These parts include object detection, object estimation and motion prediction, UAV position estimation, communication, low-level UAV control, trajectory tracking, high-level planning, and the Failure recovery and Synchronization jobs Manager (FSM), which is both the main scientific contribution of the paper, and the main contribution by the author of the thesis in this paper. The FSM is a novel approach designed to deal with the unexpected failures that occur frequently during real-world deployments of multi-robot systems.

The **experimental results** section presents the results obtained by multiple deployments of the presented system in simulated and real-world environments. Observations during the development of such a complicated system are pointed out in the following **lessons learned** section. We emphasize the significance of real-world outdoor experiments over simulations, as they may not reflect the requirements of real robotics. The last part is the **conclusions** section that summarizes the paper.

Received: 15 October 2017 | Revised: 10 April 2018 | Accepted: 12 July 2018

DOI: 10.1002/rob.21816

SYSTEMS ARTICLE

WILEY

Cooperative autonomous search, grasping, and delivering in a treasure hunt scenario by a team of unmanned aerial vehicles

Vojtěch Spurný[1] | Tomáš Báča[1] | Martin Saska[1] | Robert Pěnička[1] | Tomáš Krajník[2] | Justin Thomas[3] | Dinesh Thakur[3] | Giuseppe Loianno[4] | Vijay Kumar[3]

[1]Department of Cybernetics, Faculty of Electrical Engineering, Czech Technical University, Prague, Czech Republic

[2]Department of Computer Science, Faculty of Electrical Engineering, Czech Technical University, Prague, Czech Republic

[3]GRASP Laboratory, University of Pennsylvania, Philadelphia, Pennsylvania

[4]Department of ECE and MAE, Tandon School of Engineering, New York University, New York City, New York

Correspondence
Vojtěch Spurný, Department of Cybernetics, Faculty of Electrical Engineering, Czech Technical University, Prague, Czech Republic.
Email: vojtech.spurny@fel.cvut.cz

Funding information
České Vysoké Učení Technické v Praze, Grant/Award Number: SGS17/187/OHK3/3T/13; Office of Naval Research Global, Grant/Award Numbers: N00014-07-1-0829, N00014-14-1-0510; Army Research Laboratory, Grant/Award Number: W911NF-17-2-0181; Grantová Agentura České Republiky, Grant/Award Number: 17-16900Y; Research Center for Informatics project, Grant/Award Number: CZ.02.1.01/0.0/0.0/16_019/0000765; Khalifa University of Science, Technology and Research, Grant/Award Number: MBZIRC 2017

Abstract

This paper addresses the problem of autonomous cooperative localization, grasping and delivering of colored ferrous objects by a team of unmanned aerial vehicles (UAVs). In the proposed scenario, a team of UAVs is required to maximize the reward by collecting colored objects and delivering them to a predefined location. This task consists of several subtasks such as cooperative coverage path planning, object detection and state estimation, UAV self-localization, precise motion control, trajectory tracking, aerial grasping and dropping, and decentralized team coordination. The failure recovery and synchronization job manager is used to integrate all the presented subtasks together and also to decrease the vulnerability to individual subtask failures in real-world conditions. The whole system was developed for the Mohamed Bin Zayed International Robotics Challenge (MBZIRC) 2017, where it achieved the highest score and won Challenge No. 3—Treasure Hunt. This paper does not only contain results from the MBZIRC 2017 competition but it also evaluates the system performance in simulations and field tests that were conducted throughout the year-long development and preparations for the competition.

KEYWORDS

aerial robotics, cooperative robots, mobile manipulation, planning

1 | INTRODUCTION

Small autonomous unmanned aerial vehicles (UAVs) are widely used in numerous applications of data collection due to their potential for rapid deployment and their ability to reach locations inaccessible by ground robots. While fixed wing UAVs have the advantage of stable flight at high speeds, long range, and long flight time, rotary wing UAVs (such as the popular multirotor helicopters) benefit from their capacity for high manoeuvrability, vertical take off and landing, flight in cluttered environments in close proximity to obstacles, and hovering in a desired position in a 3D environment. The ability to precisely reach a desired 3D position and hover in place is crucial for long-term information gathering, and especially for physical interaction with objects in the workspace. Delivery applications composed of acquisition, transport, and drop-off provide an example requiring interaction with the environment during autonomous flight. This is the topic discussed in our paper.

A multiple cooperative delivery mission (called Treasure Hunt) was the most complex task in the 2017 Mohamed Bin Zayed International Robotics Challenge (MBZIRC[1]). In the competition, the

[1]http://www.mbzirc.com/ - Accessed: July 17, 2018.

delivery task was solved in its full complexity, including searching for objects with unknown positions, grasping moving objects, and cooperation among multiple UAVs working in concert. The deployment of a team of UAVs was motivated by the limited total mission time, and by including large objects with weights exceeding the maximum payload of the individual robots. In the mission, 23 objects (10 static, 10 dynamic, and 3 large) had to be localized in an outdoor arena and collected by three UAVs of limited size. While the small objects (static and dynamic) could be lifted by a single UAV, the large objects required two UAVs to transport them.

The system that exhibited the best performance among all participants in the MBZIRC competition in the Treasure Hunt challenge is presented in this paper[2]. The system design is driven by the specific task proposed and precisely specified by the organizers. The approach is tailored to provide high robustness and performance to solve the challenging task by modification of available robotic methods and designing new algorithms where necessary. Nevertheless, the proposed system is easily reusable in a large set of multi-UAV scenarios as shown in Section 1.2. The core of the system is the failure recovery and synchronization jobs manager (FSM), which is crucial for managing all subsystems and for coordinating all UAVs sharing the same workspace. The FSM is also needed to achieve the reliability required for the deployment of UAVs in real-world conditions, which requires the ability to recover from UAV failures and also from a malfunction of the localization and communication infrastructure. For example, the robots can easily collide with the objects being grasped due to a wind gust which, in combination with the ground effect, can create a hardly predictable external force on the UAV in the final phase of the approach to an object. Such a collision could result in a UAV crash, deadlock, or an overturned object. Moreover, malfunctions of UAV subsystems such as camera dropouts, incorrect rangefinder measurements, gripper failure or gripper feedback failure, and imprecise object gripping, can be expected in demanding outdoor conditions. All these eventualities need to be considered by the system to enable undisturbed operation of the remaining robots in the event of a UAV failure, limited operation of a UAV with a faulty subsystem, or an unsuccessful or interrupted grasping task. From this point of view, the proposed FSM concept can be considered as a hierarchical state machine with included synchronization and failure recovery abilities, which may be effectively reused in any complex multi-UAV task involving environment interaction.

Although the rules of the MBZIRC competition allowed the use of global navigation satellite system (GNSS) and the even more precise differential global positioning system (DGPS) for UAV localization, the availability of these systems was not guaranteed. For example, GNSS information was available only intermittently, due to interference with other transmitters located at the competition site and occlusion of satellites by the surrounding buildings and infrastructure. The provided Wi-Fi infrastructure was even less reliable and therefore the proposed FSM approach leverages the combination of different modes of the system based on the availability of Wi-Fi, GNSS, and DGPS. In addition to the FSM, a sensor fusion mechanism is presented for combining information from various onboard sensors (onboard IMU, GPS, DGPS, rangefinder, and camera) which must be considered as potentially unreliable at any time. It is vital that the UAV may continue with the task despite lacking some sensor data (e.g., precise measured altitude above ground), because the competition rules did not allow any human intervention or debugging during the trials, and which is also the case in most of the real-world UAV applications.

Another important subsystem, which is crucial in tasks requiring interaction with the environment, is relative detection and estimation of the state of the objects requiring interaction. In the presented system, the relative localization technique relies on onboard vision, since the objects in the competition were designed to support such an approach. The shape and color of the objects were specified before the mission and a color-based key was used to identify the score for collecting the particular object and to distinguish the object type. Static, dynamic, and long objects were labeled by different colors, all easily distinguishable from the background. Therefore, the vision approach is the simplest way to acquire all data required for the high-level planning (the score, type, and position estimate), and also for the visual servoing in the grasping task (precise relative positions of objects). However, any alternative relative localization system can be easily integrated based on the application. State estimation of the object is necessary mainly for dynamic objects, where a velocity estimate of the object needs to be taken into account by the UAV control modules.

Two flight behaviours are required in the Treasure Hunt task: Trajectory following and precise visual servoing. The trajectory tracking mode is used to search for the object in the environment, to approach the vicinity of the object, and to transport the object to the required location. The most important property of this controller is rapid and smooth movement along the trajectory provided by the high-level planning. The visual servoing applied in the final phase of grasping can be realized more slowly, but the requirements on precision are much higher. In the paper, we will present a novel model predictive control (MPC)-based approach that allows integration of the UAV state estimation (including external forces produced by the wind and ground effect) and target state estimation (a position and velocity estimate of the currently observed object), enabling our robots to reach the target with a maximum position error of 8 cm, which is determined by the diameter of the object and the size of the gripper.

1.1 | State-of-the-art

Rotorcraft or rotor-wing UAVs are suitable for tasks with object manipulation, due to their ability to hover on the spot. Their usage in

[2]http://mrs.felk.cvut.cz/projects/mbzirc - Accessed: July 17, 2018.

this field has already been investigated in several publications, mainly for a single UAV, in particular subparts such as gripper or manipulator design, control techniques, and object detection.

The design of a manipulator for use in industrial applications, for aerial inspection by contact, and also for aerial manipulation is described in Fumagalli, Stramigioli, and Carloni (2016). The design of a multidegree arm manipulator placed on UAVs is presented in Morton and Toro (2016) and in Korpela, Danko, and Oh (2012). The idea of using a suction-based gripper for versatile aerial grasping is presented and experimentally verified in Kessens, Thomas, Desai, and Kumar (2016). Other gripper designs are presented in Mellinger, Lindsey, Shomin, and Kumar (2011) and Pounds, Bersak, and Dollar (2011b).

A study about determining stability bounds, in which the changing mass-inertia parameters of the system due to the grasped object will not destabilize a proportional-integral-derivative flight controller for helicopters, is presented in Pounds, Bersak, and Dollar (2011a). The authors of Thomas, Loianno, Polin, Sreenath, and Kumar (2014) introduce a controller and a planner for high-speed aerial grasping, using a quadrotor UAV with a claw-like gripper. Their approach is used for grasping a cylindrical object relying on feedback from a monocular camera and an inertial measurement unit onboard the aerial robot. Images from the camera are used for computing the desired pitch angle, and the remaining axes (roll and yaw) are controlled using feedback from the vision motion capture system. In Ghadiok, Goldin, and Ren (2012), a system for autonomous grasping of objects using a monocular IR camera is introduced. Detection of the objects is based on finding an IR beacon, which has to be placed on the objects. The authors also rely only on onboard sensors, but the position and yaw estimation is computed offboard on the ground station. A methodology for controlling a multiarm manipulating aerial vehicle is presented in Orsag, Korpela, Pekala, and Oh (2013). The control of a system where the control input is generated for the UAV and the manipulator joints simultaneously is described in Heredia et al. (2014), Kamel, Comari, & Siegwart (2016), and Kannan, Quintanar-Guzman, Dentler, Olivares-Mendez, & Voos (2016). The papers (Kim, Seo, Choi, & Kim, 2016; Lippiello et al., 2016; Santamaria-Navarro, Grosch, Lippiello, Sola, & Andrade-Cetto, 2017) present a vision guidance approach using an image-based visual servo for an aerial manipulator. A method for planning a time-optimal trajectory for a quadrotor with the goal of grasping a moving target is introduced in Spica, Franchi, Oriolo, Bülthoff, and Giordano (2012). However, the solution is presented only by simulations.

Detecting and estimating the object is a challenging task that needs to be investigated for autonomous grasping. Online detection of the known object and estimation of its position using features from images are described in RamonSoria, Arrue, and Ollero (2017). Another method for onboard object extraction based on stereo vision for autonomous grasping of objects is presented in RamonSoria, Bevec, Arrue, Ude, & Ollero (2016). However, the aforementioned methods rely on stereo or depth sensors, which are not used on our UAVs. To detect the colored objects, we modified a computationally efficient method (Krajník et al., 2014), which already proved its reliability and accuracy in real-world conditions.

Ways of transporting large objects by multiple UAVs have already been investigated in Gioioso, Franchi, Salvietti, Scheggi, and Prattichizzo (2014), Mellinger, Shomin, Michael, and Kumar (2013) and Parra-Vega, Sanchez, Izaguirre, Garcia, and Ruiz-Sanchez (2013). A control scheme for cooperative simultaneous manipulation of an object by a team of UAVs is described in Parra-Vega et al. (2013). The idea of grasping and manipulating objects by a swarm of UAVs has been also studied in Gioioso et al. (2014), where the swarm is teleoperated using the free motion of a human hand. Both these works lack experimental verification, because the systems were tested only in simulations. Transport of large objects by multiple UAVs had been achieved in Mellinger et al. (2013). However, the experiments were done in an indoor environment under the Vicon[3] motion capture system.

Solutions for the Treasure Hunt scenario have already been presented by two teams participating in the MBZIRC competition which had worked on this scenario autonomously. The approach used by the team from ETH Zurich is described in Bähnemann, Schindler, Kamel, Siegwart, and Nieto (2017), and the approach used by the team from the University of Bonn is presented in Nieuwenhuisen et al. (2017). Both teams relied on an electro-permanent magnetic gripper for grasping ferrous objects, which are recognized using a color blob detection algorithm. They also used a similar approach for locating the objects. First, the arena is cooperatively searched by UAVs to create a map of the objects, and then an attempt is made to grasp and deliver each object in the map. However, the solution in Bähnemann et al. (2017) relies on a Wi-Fi communication infrastructure, and the authors do not propose any alternative in the event of communication blackout. They also do not explain how they solve the problem of multiple UAVs coordination over the drop-off zone. In Nieuwenhuisen et al. (2017), the authors mention a conservative solution for a disturbed communication network. However, this solution is not explained in detail, and therefore, their approach cannot be directly replicated and evaluated. Furthermore, their controller does not compensate for external factors such as wind, which is a common disturbance in an outdoor environment.

1.2 | Contribution

The contribution of this paper correlates directly with the expected contribution of the MBZIRC challenge. A board of respected scientists[4] from leading robotic groups worldwide selected the Treasure Hunt scenario as the most challenging task in the MBZIRC event for numerous reasons. This scenario extends state-of-the-art systems in various ways: Deployment of multiple UAVs in the same outdoor workspace, multirobot scanning of the environment with no prior information on the position of objects, online distribution of tasks to UAVs based on the obtained information, and physical interaction with the environment. Indeed, physical interaction of UAVs with objects in an unknown outdoor environment, especially

[3]http://www.vicon.com/ - Accessed: July 17, 2018

[4]http://www.mbzirc.com/committee - Accessed: July 17, 2018

cooperatively (some objects require the cooperation of multiple UAVs), is a challenging and innovative task, mainly if it must be solved in demanding windy environments, such as the MBZIRC 2017 venue in Abu Dhabi. The strong wind gusts present in the location between the coast and the desert significantly influence the precision and the stability of the UAV controllers, particularly in the final phase of object grasping, where they are combined with the ground effect. Further, the light conditions (e.g., the strong and variable sunshine) make the vision task more complicated than in a laboratory environment. The multirobot aspect requires rapid communication and coordination of UAVs, which seemed to be a bottleneck for the approaches presented by most of the other teams. Our solution to the challenges caused by unreliable communication is also a contribution to robotic research.

The overall contribution of our paper goes beyond the MBZIRC challenge, as it contains a comprehensive description of all components of the system that can be used in various collaborative multi-UAV missions, including physical interaction of robots and the environment. Although the system is primarily designed for outdoor deployment with a GNSS signal available, it can be used in GNSS-denied conditions with only a slight modification, since object grasping is realised by visual servoing, which relies on relative localisation only. Besides object grasping and delivery tasks, the system has been successfully deployed in numerous multi-UAV applications, including detection of sources of radiation and electromagnetic fields (Saska, 2017), inspection and documentation of historical sites (Saska, Kratky, Spurny, & Baca, 2017), reconnaissance and surveillance missions (Pěnička, Faigl, Váňa, & Saska, 2017; Pěnička, Saska, Reymann, & Lacroix, 2017), etc.

Another contribution of this paper for the robotic community is based on the fact that the next MBZIRC event intends to build on the achievements of MBZIRC 2017, and to propose even more challenging tasks that are beyond the current state-of-the-art in robotics. Although 143 teams applied to participate in the 2017 contest, including the best robotic labs worldwide, only four groups were able to grasp at least one object autonomously during the competition. To maximize the impact of future MBZIRC events and to encourage more competition, which will again push the limits of robotic systems, it is necessary for more teams to succeed in solving the challenging scenarios. A logical starting point is to use, or at least be inspired by, the approach that demonstrated the best performance in the 2017 MBZIRC, which is presented in this paper. Moreover, we would like to share and highlight the parts of the system and the phases in its development that brought added value in comparison with the systems of our competitors. Our experience and our solutions to the proposed challenges should be beneficial in further MBZIRC contests, in other robotic competitions, and also for the design of autonomous UAV systems for deployment in emergency applications. The rules of the competition forced teams to design a system for immediate deployment (the preparation time was only 20 min for the multi-UAV challenge) and for operation within a given time, without the option of postponing the start of the mission. This contrasts with most robotic experiments presented in the literature, where only successful trials and demos are presented. Short preparation time and a successful start on demand, without the possibility of repeated trials, are required by industry and in emergency applications, and the MBZIRC competition was designed to force teams to achieve these requirements.

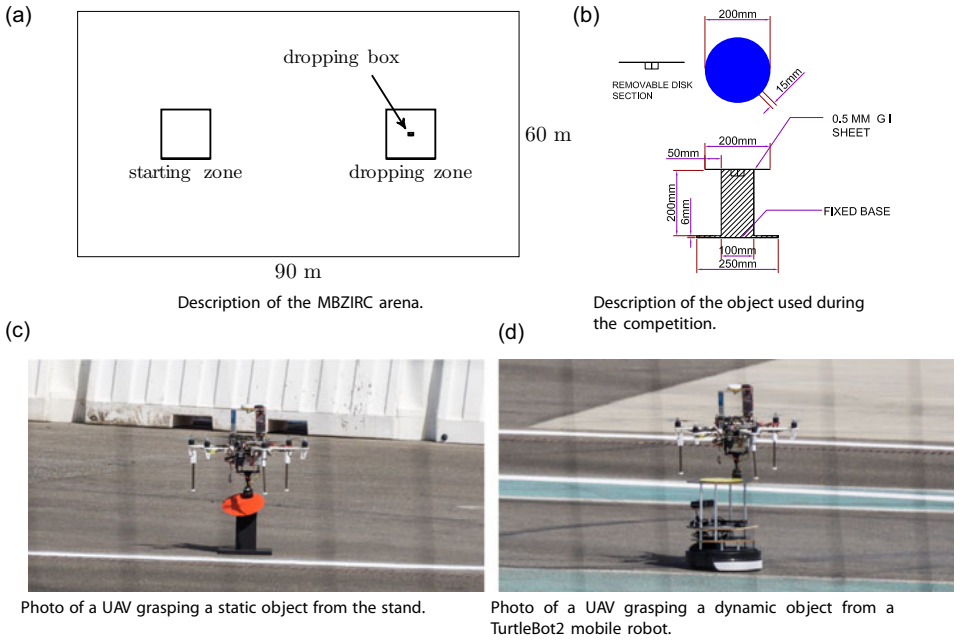
1.3 | Problem statement

In the MBZIRC 2017 Treasure Hunt challenge, three UAVs (with a maximum size of 120 cm × 120 cm × 50 cm) must locate, grasp, deliver, and drop a set of objects into a given box within 20 min. The set should contain 10 moving and 10 stationary small objects, as well as three stationary large objects, all of which are randomly placed inside the arena. The small objects were approximately 0.370 kg ferrous disks on a stationary stand or moving TurtleBot2 robot, as shown in Figure 1b–d. Different colors of the objects—green, blue, and red for the static objects, and yellow for the dynamic objects—were associated with different scores, one, two, three, and five points, respectively. The nonstationary objects were moving at random velocities not exceeding 5 km/hr. Three large orange objects not exceeding 200 cm in length, and not exceeding 2 kg in weight, were valued at ten points each on successful transport and delivery by at least two cooperating UAVs into the dropping zone depicted in Figure 1a. If a large object was moved into the dropping zone by a single robot, the team obtained five points. The small objects could be grasped by a single UAV and dropped into a box placed inside the dropping zone. The objects could be picked up by a magnetic gripper, a suction gripper, or another device carried onboard the UAVs. Before the start of each trial, the three UAVs had to be in the start location.

2 | HARDWARE

The specifications of the MBZIRC challenge described above influence the decision on which UAV platform to use. Our intention was to reuse the platforms and the entire system in our follow-up research, and to achieve simple replicability of the system in the future. Therefore, we tried to maximize the use of commercially available off-the-shelf UAV components, and only a few 3D printed specialized tools (sensor holders and the gripper). This approach reduced development time, increased reliability, and now enables our system to be used by other universities with a minimum overhead for technology transfer. It also increases the impact of this paper, which can be considered as a comprehensive manual for building a robust multi-UAV system, even for research groups without any experience with UAVs.

The proposed UAV platform is a complex system composed of integrated active members, computational resources, and sensor modules, shown in a schematic view of the system in Figure 2. The main structure of each UAV consists of a DJI hexacopter F550 frame and E310 DJI motors. This choice satisfies the size limitations of the MBZIRC event, the flight time, and the payload capability that is necessary for additional sensors, and also for carrying the objects. The



Description of the MBZIRC arena.

Description of the object used during the competition.

Photo of a UAV grasping a static object from the stand.

Photo of a UAV grasping a dynamic object from a TurtleBot2 mobile robot.

FIGURE 1 Description of the MBZIRC 2017 competition. For more information, visit http://www.mbzirc.com. UAV: unmanned aerial vehicle [Color figure can be viewed at wileyonlinelibrary.com]

system is controlled at the lowest level by a PixHawk flight controller (Meier et al., 2012) that contains a set of sensors, such as accelerometers, gyroscopes, and magnetometers, which are necessary for stable UAV flight. The open-hardware and open-software architecture is advantageous for the MBZIRC competition, and also for research on multirobot systems. An Intel NUC-i7 PC provides sufficient computation power to solve all the required onboard image processing tasks, and also UAV coordination, state estimation, and motion planning in the complex Treasure Hunt challenge. Transport of messages between the onboard PC and PixHawk autopilot is performed over a serial line using MAVlink protocol. Communication between the UAVs, which is important for their coordination, is provided by the Wi-Fi module embedded in the PC. A high-resolution Mobius ActionCam (2018) camera is used for object detection, and for relative visual localization.

The rules of the competition allowed the use of GNSS and even more precise navigation systems for localization. To maximize the accuracy and to increase reliability, our system uses a combination of the real-time kinematic (RTK) satellite, which enhances the precision of position data derived from satellite-based positioning systems (e.g. GPS, GLONASS, Galileo, and BeiDou), and a classical GNSS module attached to the PixHawk controller. Information on the position is provided in the RTK system by a PRECIS-BX305

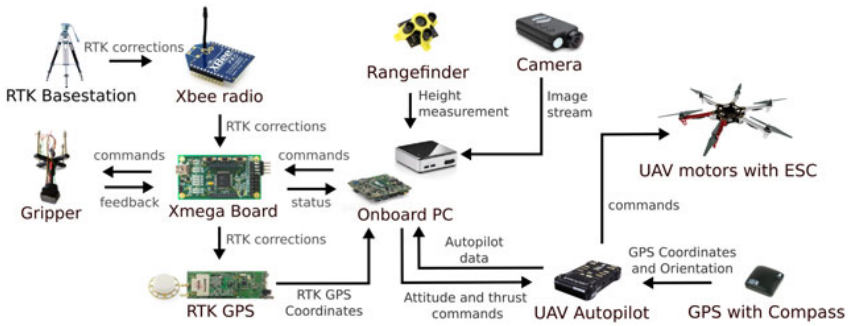


FIGURE 2 Description of components in our UAV platform. ESC: electronic speed controller; GPS: global positioning system; RTK: real-time kinematic; UAV: unmanned aerial vehicle [Color figure can be viewed at wileyonlinelibrary.com]

GNSS RTK BOARD (GPS L1L2/GLONASS G1/BEIDOU B1B3) (Tersus-GNSS, 2017), with accuracy of 10 mm ± 1 parts per million (ppm) horizontally and 15 mm ± 1 ppm vertically when the RTK device is in the most accurate state, RTK FIX. This RTK system requires a stationary GNSS receiver, called RTK basestation, which is placed on a known location. The RTK basestation then broadcasts its position and measurements from all visible satellites (RTK corrections) to the UAVs using XBee Pro radio modules (Digi International, 2017). A custom board was designed to provide communication of the XBee module with the RTK device.

In principle, the vertical position (altitude) provided by the RTK GPS is measured above the mean sea level. However, the UAV does not have any information about the ground-level profile or the distance to the objects that are to be grasped, based on the GPS. This information is obtained using the onboard *TeraRanger One* laser rangefinder, which is mounted face-down and is connected directly to the onboard PC, where its data are filtered and used for precise height control. Finally, the objects are grasped using an OpenGrab EPM v3 electropermanent magnet, which combines the advantages of electro and permanent magnets and creates a very strong magnetic contact with ferrous objects (NicaDrone, 2017). Our custom board (previously mentioned for managing communication from the XBee module into the RTK device) also provides a low-level interface between the main computer and the gripper.

3 | SOFTWARE SYSTEM STRUCTURE

The proposed solution relies on the robot operating system (ROS), which is an open-source set of software libraries and tools commonly used in the robotic community. Using ROS, the complex MBZIRC tasks can easily be divided into smaller subtasks (nodes). This also improves and clarifies the structure of the proposed solution. Furthermore, the Gazebo robotic simulator can be used for simulation in the loop, together with firmware from PixHawk, which provides a very realistic testbed and significantly simplifies testing of the whole system. Using this realistic simulator, hardware experiments could be carried out in a shorter time and in a safer way than if direct HW is used. Because changes were double-checked in the simulator, we did not experience any serious crash during more than 1 year of intensive preparation for the MBZIRC event.

In this section, the subcomponents of the proposed system are described. The first two parts explain object detection, object estimation, and motion prediction for dynamic objects. In the next subsection, the estimation of the UAV state from all available sensors is introduced, followed by details on communication in the multirobot network. Further, the nonlinear controller used for UAV control is explained, together with the novel MPC-based approach used for online design of a feasible and smooth reference for the nonlinear controller. This is followed by details of high-level planning built upon MPC-based trajectory tracking, which is used for UAV coordination and collision avoidance when the same workspace is shared. Lastly, the FSM, which is crucial for managing all subsystems and for coordinating all UAVs sharing the same workspace, is described. All these subcomponents are executed on the onboard PC Intel NUC-i7.

3.1 | Object detection

Since the camera that is used to detect the colored objects has a rolling shutter, vibrations induced by the drone motors cause the acquired images to be subject to a specific 'jelly' or 'wobble' effect, which makes the use of geometry-based methods for object detection (e.g., the Hough transform) problematic (Afolabi, Man, Liang, Guan, & Krilavičius, 2015; see Figure 3). We therefore designed a computationally efficient ellipse detection algorithm, which relies on the use of statistics that are robust to this type of noise (Krajník et al., 2014). However, the original method described in Krajník et al. (2014), which used adaptive thresholding to detect black-and-white patterns, had to be extended to process the color information.

Since the perceived colors are influenced by the light conditions, and the exact colors of the objects were not known until the actual contest, we created a semiautomatic autocalibration method that can learn a Gaussian-mixture-based model (GMM) of each color during a short hover over the objects. Once the GMMs are learned, they are used to create an RGB color map, which allows the image pixels to be classified rapidly into object candidates and the background.

The color map is then used in the method (Krajník et al., 2014), which searches for continuous segments of object-colored pixels,

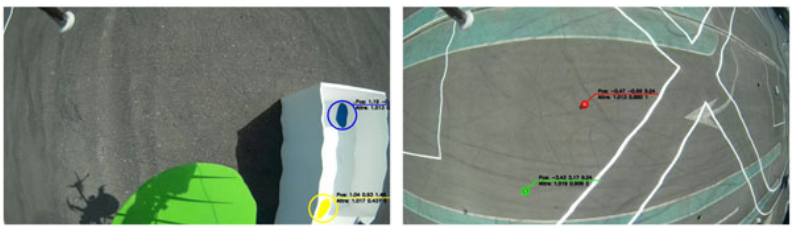


FIGURE 3 Object detection in onboard camera images affected by the 'jelly' or 'wobble' effect, which deforms lines (left image), as well as circular and square objects (right image). The detection results indicate the 3D relative position (top line) and attributes like roundness, eccentricity and type (1,2,3 for red, green, and blue static objects and 5 for the yellow moving object) [Color figure can be viewed at wileyonlinelibrary.com]

establishes their bounding box, the number of pixels, the centroid, convexity, and compactness and uses these statistics to reject segments that cannot correspond to circular objects. Then, using the known object size and camera parameters, the method calculates the relative 3D position of the object. This position is then transformed to a global 3D coordinate frame, and objects that do not appear to be close to the ground plane are rejected as false positives. Finally, global 3D positions of the detected objects are forwarded to a mapping module, which integrates multiple detections of the objects into a single 3D representation, which is then used by the planning system.

The performance of the method during tests and in the contest itself indicated computational efficiency and robustness to changing illumination, which was one the key factors in the robustness of the entire system used in the MBZIRC competition.

3.2 | Object estimation and motion prediction

Localization of targets with onboard cameras tends to provide data that are inherently embedded with flaws. The data may be skewed by phenomena such as *signal noise*, *false positive detections*, *irregular detection rate*, *data blackouts*, etc. These issues can hardly be mitigated during the detection, and some of them (e.g., data blackouts) also depend on the external environment. Moreover, several moving targets appear in the MBZIRC challenge and so estimates of unobserved states such as velocities and heading may help to follow their position precisely. This leads to a need to filter the detected position of the targets. We also required the filtration system to be capable of sorting out measurements belonging to targets that have been marked as *unreliable*, for example, due to data blackout being too frequent. Another requirement comes from the multirobot nature of the task. A UAV should share information about parts of the map that are currently occupied. Targets in those areas should then be filtered out in other UAVs to prevent unrequired grasping of the same target by multiple UAVs.

In the event that there is a single target in the field of view (FOV) of the UAV, an Unscented Kalman Filter (UKF) is used as a filter and as a predictor in conjunction with the car-like motion model

$$\begin{aligned}
\mathbf{x}^o_{[n+1]} &= \mathbf{x}^o_{[n]} + \dot{\mathbf{x}}^o_{[n]}\Delta t,\\
\dot{\mathbf{x}}^o_{[n+1]} &= \begin{pmatrix}\cos\phi_{[n+1]}\\ \sin\phi_{[n+1]}\end{pmatrix} v_{[n+1]},\\
\phi_{[n+1]} &= \phi_{[n]} + \dot{\phi}_{[n]}\Delta t,\\
\dot{\phi}_{[n+1]} &= K_{[n]} v_{[n]},\\
v_{[n+1]} &= v_{[n]} + a_{[n]}\Delta t,\\
K_{[n+1]} &= K_{[n]} + \dot{K}_{[n]}\Delta t,
\end{aligned} \tag{1}$$

where $\mathbf{x}^o_{[n]} = (x, y)^T_{[n]}$ is the position of the object in the global coordinate system, $\phi_{[n]}$ is its heading, $K_{[n]}$ is the curvature of its turn, $v_{[n]}$ is its scalar velocity, $a_{[n]}$ is its scalar acceleration, and Δt is the time difference. An estimate of the target heading allows its motion to be tracked, while the onboard camera is oriented with its wider FOV in favor of detecting sudden changes of the object's heading.

However, real-world scenarios might contain several objects in the FOV, while some of them are moving. In that case, the UAV needs to track a particular object independently of the movement of all the objects in the scene. This requires a local map of the objects to be actively maintained. Our map model was based on Equation (1) for an arbitrary number of independent objects. Another state has been included to cover the type of the object (its color and whether it is moving) as well as the time of its last update and whether it is currently active. Manipulation of the objects in the map obeys the following principles:

- An object that has not been seen for more than 5 s is *deactivated*. Deactivated objects stay in the map, but their movement is no longer predicted by the UKF.
- Objects that are deactivated for more than 3 s are deleted from the map.
- Measurements from the *object detector* (Section 3.1) are paired with objects in the map using min-distance bipartite graph matching, constrained by the color of the objects.
- Objects located outside of the competition arena or in any of the locally banned areas (near the dropping zone or around other UAVs) are deleted from the map, and new measurements in these areas are discarded.

Additionally, it can be anticipated that grasping attempts may not be successful at all times. The filter allows a temporarily ban on an area around a particular object, to avoid deadlock in the grasping state machine. Such a ban is valid for 30 s in a radius of 4 m around the object.

3.3 | UAV position estimation

Automatic control of UAVs relies on estimates of the states of the UAV dynamical system. Namely, knowledge of position and velocity (both vertical and horizontal) is required to coordinate the movement for precise picking up and delivery of the object. Our platform is equipped with several independent sources of information, which are fused to obtain a single, reliable and smooth estimate of the UAV pose. An important requirement is to ensure smoothness of the resulting signal, since the $SO(3)$ state feedback is sensitive to noise.

The main source of data for both the vertical and the horizontal axes is the PixHawk flight controller. Its extended Kalman filter fuses present-day inertial sensors—a three-axis accelerometer and a gyroscope with an altitude pressure sensor and a GPS receiver. Although the aircraft is already capable of autonomous flight with this off-the-shelf setup, we make use of other sensors to provide more precise localization and thus better precision of object manipulation.

3.3.1 | Horizontal position estimation

The position estimates in the lateral axes are based on the estimate provided by PixHawk, namely positions $\mathbf{x}^p$, and velocities $\dot{\mathbf{x}}^p$. Although the precision of the estimates may be satisfactory locally for short time intervals, they are prone to significant drift in time spans of minutes.

To correct this drift, and thus to ensure repeatability of the experiments and, for example, locating the dropping zone, the horizontal position from PixHawk is corrected by differential RTK GPS. Position measurements from the RTK GPS receiver are fused using the linear Kalman filter with the model

$$\mathrm{A} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}, \mathrm{B} = \begin{pmatrix} \Delta t \\ \Delta t \end{pmatrix}, \tag{2}$$

where $\mathbf{x}^e_{[n+1]} = \mathbf{A}\mathbf{x}^e_{[n]} + \mathbf{B}\mathbf{u}_{[n]}$ is the linear system equation, $\mathbf{x}^e_{[n]} = (x, y)^T_{[n]}$ is the state vector finally used for control, and $\mathbf{u}_{[n]}$ is the system input. According to our experience,

$$\mathrm{x}^p_{[0]} + \sum_{n=0}^{k} \dot{\mathrm{x}}^p_{[n]} \Delta t_{[n]} = \mathrm{x}^p_{[k]}, \quad \forall\, k \in \mathbb{N} \tag{3}$$

does not hold for the position and velocity estimate provided by PixHawk. This is a very useful observation for somebody building a fully autonomous UAV system using an off-the-shelf controller. The input vector $\mathbf{u}$ consists of velocities obtained as differentiated positions $\mathbf{x}^p$ (later integrated by the filter), which ensures that the proposed filter does not introduce any drift into the resulting estimate when no RTK GPS corrections are received. In situations when the position is not being corrected, the resulting estimate follows the same relative state trajectory as $\mathbf{x}^p$, just shifted according to the latest correction. The final tuning of the filter resulted in process covariance $\mathbf{Q} = \mathrm{diag}(1, 1)$ and measurement covariance $\mathbf{R} = \mathrm{diag}(10e3, 10e3)$. Moreover, the RTK GPS corrections were saturated to ever impose maximally 0.25 m difference from the internal state of the filter. Such technique limits sudden changes of the estimated position, which was necessary for safety of the flight.

The multirobotic scenario requires a coordinate space to be shared among all three UAVs. The base of our Cartesian system is set to predefined GPS coordinates and its orientation is according to the East-North-Up convention. Therefore, the first, second, and third axis point to the east, north and upwards, respectively. A point of origin is measured using the RTK GPS, to which all independent coordinate systems of all UAVs are then shifted after each of them is powered up. The common base station of the differential RTK GPS then ensures that all UAV estimates are corrected to lie within the same global coordinates.

3.3.2 | Vertical position estimation

In contrast with the horizontal position, estimates of the height rely much less on PixHawk. The linear Kalman filter for the vertical axis also uses the differentiated PixHawk height in the same manner as the horizontal axis. However, height corrections come not only from the differential RTK GPS, but also from the down-facing TeraRanger rangefinder and the object detector, which is able to provide an estimate of the relative distance, when flying above an object. The estimator provides an option to switch between these sources of data, depending on the current task and the circumstances.

It is feasible to correct the height using the TeraRanger rangefinder, when flying above uneven ground, but it cannot be used reliably when the down-facing sensor is obstructed, for example, when carrying an object, or when there might be a foreign object on the ground, namely the dropping box. RTK GPS can provide precise relative height measurements, but only when RTK FIX has been established. This depends on the strength of the GNSS signal and on the quality of the communication link between the base station and the UAV. Finally, correcting the altitude using data from the *object detector* may bring in unexpected steps in the signal due to false-positive detections. Since none of the additional sources is completely reliable, we implemented a safety mechanism for detecting anomalies, which can toggle off any of the above-mentioned sensors from being fused.

3.4 | Communication between UAVs

In multirobot systems, reliable communication is required mainly if there is a need for direct cooperation between multiple autonomous vehicles, as in the case when large objects are to be carried cooperatively. However, a reliable communication channel is a crucial tool even for coordinating the UAV team sharing the same workspace for grasping small objects individually, as was demonstrated in the MBZIRC competition. The rules of the MBZIRC event specified that all teams are obliged to share the same 5 GHz Wi-Fi network, the reliability of which was influenced by interference occurring during transmission. This may easily lead to packet loss, which can interrupt the connection. Decreased reliability of the communication link during the entire mission is not limited to the MBZIRC case. It is a typical feature of most UAV applications in demanding outdoor conditions. The MBZIRC contest therefore provided an interesting and realistic evaluation scenario for multi-UAV systems, in which it cannot be assumed that a complete communication network is available at all times. In our opinion, our system achieved significantly better performance in the multi-UAV scenario than the other teams, due to the following strategy. We attempted to maximize utilization of the communication channel, if it was available, to achieve optimal behavior of the system. However, it was important to be able to degrade into a system not relying on the communication infrastructure at all. This was done at the cost of decreased performance, but our system still provided safe flight operation of multiple UAVs solving the given task. A smooth and possibly repeated transition between the optimal behavior relying on communication and the nonoptimal but safe and still working system without communication, and back, is provided by the FSM approach described in Section 3.8.

The software part responsible for managing communication between UAVs is based on the ROS master within the ROS network. To increase the robustness of the communication net in the event of a failure of the robot that is the leader in the ROS master scheme, the proposed method relies on multiple independent ROS masters assigned to each of the UAVs. The ROS package multimaster_fkie (Tiderko, 2017) is used to maintain communication between these ROS masters. This package offers a set of nodes to establish and

manage a multimaster network, which is necessary for such tasks with the team of UAVs in the event of an unreliable communication infrastructure.

To reduce the load of the communication channels managed by the ROS master network, only selected information (topics) are exchanged between the team members:

- the actual position of the UAV in the global coordination system,
- the actual state of the high-level state machine being part of the FSM,
- the estimated position of the object during grasping,
- the planned trajectory.

These topics are used in nodes for proactive collision-free planning, fail-safe reactive collision avoidance, and object estimation. The bandwidth of the Wi-Fi network necessary for transmission of all mentioned information for a single UAV is approximately 10 kB/s.

3.5 | Low-level UAV control

The position controller uses the estimated state as feedback to follow the trajectories given as an output of the high-level trajectory planner. In many previous works, a backstepping approach is used for UAV control, because the attitude dynamics can be assumed to be faster than the dynamics governing the position, so linearized controllers are used for both loops (Herissé, Hamel, Mahony, & Russotto, 2012; Mellinger et al., 2013; Weiss, Scaramuzza, & Siegwart, 2011). However, we need the system to be capable of large deviations from the hover configuration during operations like fast mapping of objects, or for strong wind compensation. We therefore use a nonlinear controller. Let us consider an inertial reference frame denoted by $[\mathbf{e}_1, \quad \mathbf{e}_2, \quad \mathbf{e}_3]$ and a body reference frame centered in the center of mass of the vehicle with an orientation denoted by $\mathbf{R} = [\mathbf{b}_1, \quad \mathbf{b}_2, \quad \mathbf{b}_3]$, where $\mathbf{R} \in SO(3)$. The dynamic model of the vehicle can be expressed as

$$\begin{aligned} \dot{\mathbf{x}} &= \mathbf{v}, \\ m\dot{\mathbf{v}} &= f\mathbf{R}\mathbf{e}_3 + mg\mathbf{e}_3, \\ \dot{\mathbf{R}} &= \mathbf{R}\hat{\boldsymbol{\Omega}}, \\ \mathbf{J}\dot{\boldsymbol{\Omega}} + \boldsymbol{\Omega} \times \mathbf{J}\boldsymbol{\Omega} &= \mathbf{M}, \end{aligned} \tag{4}$$

where $\mathbf{x} \in \mathbb{R}^3$ is the Cartesian position of the vehicle expressed in the inertial frame, $\mathbf{v} \in \mathbb{R}^3$ is the velocity of the vehicle in the inertial frame, $m \in \mathbb{R}$ is the mass, $f \in \mathbb{R}$ is the net thrust, $\boldsymbol{\Omega} \in \mathbb{R}^3$ is the angular velocity in the body-fixed frame, and $\mathbf{J} \in \mathbf{R}^{3\times3}$ is the inertia matrix with respect to the body frame. The hat symbol $\hat{\cdot}$ denotes the skew-symmetry operator according to $\hat{\mathbf{x}}\mathbf{y} = \mathbf{x} \times \mathbf{y}$ for all $\mathbf{x}, \mathbf{y} \in \mathbb{R}^3$, g is the standard gravitational acceleration, and $\mathbf{e}_3 = [0 \quad 0 \quad 1]^\mathsf{T}$. The total moment $\mathbf{M} \in \mathbb{R}^3$, with $\mathbf{M} = [M_1 \quad M_2 \quad M_3]^\mathsf{T}$, along all axes of the body-fixed frame and the thrust $\tau \in \mathbb{R}$ are control inputs of the plant. The dynamics of the rotors and propellers are neglected, and it is assumed that the force of each propeller is directly controlled. The total thrust, $f = \sum_{j=1}^{6} f_j$, acts in the direction of the z-axis of the body-fixed frame, which is orthogonal to the plane defined by the centers of the six propellers. The relationship between a single motor thrust f_j, the net thrust f, and the moments $\mathbf{M}$ can be written as

$$\begin{bmatrix} f \\ M_1 \\ M_2 \\ M_3 \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 \\ sd & 1 & sd & -sd & -1 & -sd \\ -cd & 0 & cd & cd & 0 & -cd \\ -1 & 1 & -1 & 1 & -1 & 1 \end{bmatrix} \begin{bmatrix} f_1 \\ f_2 \\ f_3 \\ f_4 \\ f_5 \\ f_6 \end{bmatrix}, \tag{5}$$

where $c = \cos(30^\circ)$, $s = \sin(30^\circ)$, and d is the distance from the center of mass to the center of each rotor in the $\mathbf{b}_1, \mathbf{b}_2$ plane. For nonzero values of d, Equation (5) can be inverted using the right pseudoinverse.

For control, we build on the work in Lee, Leok and McClamroch (2013) and in Mellinger and Kumar (2011) with control inputs $f \in \mathbb{R}$ and $\mathbf{M} \in \mathbb{R}^3$ chosen as

$$\mathbf{M} = -k_R\mathbf{e}_R - k_\Omega\mathbf{e}_\Omega + \boldsymbol{\Omega} \times J\boldsymbol{\Omega} - J(\hat{\boldsymbol{\Omega}}\mathbf{R}^T\mathbf{R}_c\boldsymbol{\Omega}_c - \mathbf{R}^T\mathbf{R}_c\dot{\boldsymbol{\Omega}}_c), \tag{6}$$

$$\begin{aligned} f = -\Big(&-k_x\mathbf{e}_x - k_{ib}\mathbf{R}\int_0^t \mathbf{R}(\tau)^T\mathbf{e}_x d\tau - k_{iw}\int_0^t \mathbf{e}_x d\tau - k_v\mathbf{e}_v - mg\mathbf{e}_3 \\ &+ m\ddot{\mathbf{x}}_d\Big) \cdot \mathbf{R}\mathbf{e}_3 = \mathbf{f} \cdot \mathbf{R}\mathbf{e}_3, \end{aligned} \tag{7}$$

with $\ddot{\mathbf{x}}_d$ the desired acceleration, and k_{iw}, k_{ib}, k_x, k_v, k_R, k_Ω positive definite terms. We extend the referenced controllers by including two integral terms which accumulate the error in the body frame and in the world frame, respectively. We include both terms to provide the opportunity to capture external disturbances (e.g., wind) separately from internal disturbances (e.g., an inefficient prop or a payload imbalance), particularly when the vehicle is permitted to yaw or rotate about the vertical axis. The thrust and the moments are then converted to motor rates according to the characteristic of the proposed vehicle. Subscript C denotes a commanded value, and $R_C = [\mathbf{b}_{1,C}, \quad \mathbf{b}_{2,C}, \quad \mathbf{b}_{3,C}]$ is calculated as

$$\begin{aligned} &\mathbf{b}_{2,des} = [-\sin\psi_{des}, \quad \cos\psi_{des}, \quad 0]^\mathsf{T}, \qquad \mathbf{b}_{3,C} = \frac{\mathbf{f}}{||\mathbf{f}||}, \\ &\mathbf{b}_{1,C} = \frac{\mathbf{b}_{2,des} \times \mathbf{b}_3}{||\mathbf{b}_{2,des} \times \mathbf{b}_3||}, \qquad \mathbf{b}_{2,C} = \mathbf{b}_3 \times \mathbf{b}_1. \end{aligned} \tag{8}$$

Note that here we have to define $\mathbf{b}_{2,des}$ based on the yaw, instead of defining $\mathbf{b}_{1,des}$ as it was defined in Mellinger and Kumar, 2011, due to a different Euler angle convention (we use the ZYX convention instead of ZXY). The quantities

$$\begin{aligned} \mathbf{e}_R = \tfrac{1}{2}(\mathbf{R}_C{}^\top\mathbf{R} - \mathbf{R}^\top\mathbf{R}_C)^\vee, \quad \mathbf{e}_\Omega &= \boldsymbol{\Omega} - \mathbf{R}^\top\mathbf{R}_C\boldsymbol{\Omega}_C, \\ \mathbf{e}_x = \mathbf{x} - \mathbf{x}_d, \quad \mathbf{e}_v &= \dot{\mathbf{x}} - \dot{\mathbf{x}}_d, \end{aligned} \tag{9}$$

represent the orientation, the angular rate errors, and the translation errors, respectively. The symbol $.^\vee$ represents the *vee* map $\mathrm{so}(3) \to \mathbb{R}^3$. If the initial attitude error is less than 90°, the zero equilibrium of the tracking error is exponentially stable, that is, $[\mathbf{e}_x^\mathsf{T} \quad \mathbf{e}_v^\mathsf{T} \quad \mathbf{e}_\Omega^\mathsf{T} \quad \mathbf{e}_R^\mathsf{T}]^\mathsf{T} \equiv [\mathbf{0}^\mathsf{T} \quad \mathbf{0}^\mathsf{T} \quad \mathbf{0}^\mathsf{T} \quad \mathbf{0}^\mathsf{T}]^\mathsf{T}$.

3.6 | Trajectory tracking

The state feedback, described in Section 3.5, which provides precise position and velocity control, requires a smooth and feasible reference. The reference consists of all states of the translational dynamics—position, velocity, and acceleration—and is provided at 100 Hz, the same rate as the resulting control signal. There are various ways to create the reference. Typically, thanks to the differential flatness of the UAV dynamical system, a QP optimization can be solved to find a polynomial given the initial and final state conditions (Mellinger & Kumar, 2011), which can then be sampled to create the reference. In our case, we chose to generate the reference using a MPC approach. MPC ensures that the resulting trajectory satisfies a given model as well as the dynamical constraints, which are imposed on the model. As it optimizes control actions over a prediction horizon, it can react adequately to unfeasible changes in the reference trajectory, and can also create proper feed-forward proactions to minimize the control error in the future.

The MPC tracker uses a QP formulation of a minimal sum-of-squares problem, where the optimal control action $\mathbf{u}$ is found for a future prediction horizon of states $\mathbf{x}_{[n]} = (x, \dot{x}, \ddot{x}, y, \dot{y}, \ddot{y}, z, \dot{z}, \ddot{z})^T_{[n]}$ by minimizing the function

$$\begin{aligned} V(\mathbf{x}_{[0,\dots,m-1]}, \mathbf{u}_{[0,\dots,m-1]}) &= \frac{1}{2}\sum_{i=1}^{m-1}\left(\mathbf{e}^T_{[i]}\mathbf{Q}\mathbf{e}_{[i]} + \mathbf{u}^T_{[i]}\mathbf{P}\mathbf{u}_{[i]}\right), \\ \text{s.t.}\quad & \mathbf{x}_{[0,\dots,m-1]} \geq \mathbf{x}_L, \\ & \mathbf{x}_{[0,\dots,m-1]} \leq \mathbf{x}_U, \end{aligned} \tag{10}$$

where $\mathbf{e}_{[n]} = \mathbf{x}_{[n]} - \tilde{\mathbf{x}}_{[n]}$ is the control error, $\tilde{\mathbf{x}}_{[n]}$ is the setpoint for the MPC, m is the length of the prediction horizon, and $\mathbf{x}_L$ and $\mathbf{x}_K$ represent box constraints on states. The control error $\mathbf{e}_{[n]}$ requires the formation of a general prediction of $\mathbf{x}_{[n]}$, which was described previously in Baca, Loianno, and Saska (2016). In our case, the optimized control action is not directly used to control the real UAV. Instead, it controls a model of the UAV translational dynamics in real-time simulation. States of the simulated model are then sampled at 100 Hz to create the reference for the state feedback. This is a novel approach in UAV control, where benefits of both nonlinear control and linear MPC are used together.

An important notion is the difference between the trajectory setpoint $\tilde{\mathbf{x}}$ and the reference, which is generated by the MPC tracker. The trajectory setpoint $\tilde{\mathbf{x}}$ is provided by an operator or a program. No requirements are imposed on $\tilde{\mathbf{x}}$. In contrast, the reference produced by the MPC Tracker is feasible, satisfies the UAV dynamics and state constraints, and serves as a control reference for the $SO(3)$ state feedback.

The simulated model is a linear time-invariant system covering the third-order translational dynamics of the UAV with sampling of $\Delta t = 0.01$ s. In our MPC formulation, Δt is different for the first iteration ($\Delta t = 0.01$ s) and for all the other iterations ($\Delta t = 0.2$ s). This allows smooth control of the simulation, if the MPC is executed at 100 Hz, while there is a relatively sparse distribution of further states, which allows us to have a much longer prediction horizon than there would normally be with Δt being constant. As in traditional MPC, only the control action in the step is used to control the model in the simulation. In the meantime, a new instance of the optimization task is formulated, starting from new initial conditions, which results in a fresh control action for the next step. This method is valid only if the MPC can be solved repeatedly within 0.01 s.

The penalization parameters $\mathbf{Q}$ and $\mathbf{P}$ in Equation (10) were found empirically. As in our previous work (Baca et al., 2016), we used the *move blocking* technique to effectively prolong the prediction horizon while maintaining the computational complexity. The particular control action distribution for the MBZIRC competition was as

$$\mathbf{U} = (1 \quad 1 \quad 1 \quad 1 \quad 1 \quad 5 \quad 5 \quad 5 \quad 5 \quad 5 \quad 10), \tag{11}$$

which results in an 8-s prediction horizon with only 33 variables in the optimization task.

Creating the control reference for the state feedback with MPC has several advantages over conventional solutions. It produces a reference that is feasible according to the specified model, which makes it safe to execute. If the setpoint for MPC is not feasible, the resulting reference is feasible with respect to Equation (10). The inherent predictive nature of MPC provides trajectory tracking optimizing actions over the future, which makes it ideal for tracking moving targets, such as the moving objects in the competition.

As defined in Equation (10), MPC handles state constraints as linear constraints. We impose maximum acceleration and velocity box constraints on the UAV to ensure safe and feasible resulting trajectories. The optimization being solved lies in the family of linearly constrained quadratic programming, which acquires a global optimum in a convex polytope. A custom solver, based on a sequential closed-form solution, has been implemented to ensure guaranteed real-time performance.

MPC-based trajectory tracking operates in two modes, as follows. The first simple positioning mode, used mainly for short distance position changes, applies either relative or absolute position commands, and tries to reach a given position in the fastest way with respect to the MPC scheme. The second trajectory-following mode used by high-level trajectory planning (Section 3.7) uses a precomputed path plan, and tries to precisely track the trajectory while respecting the plan waypoints schedule, which is crucial for multirobot collision-free operation.

Having the predictions of the future movement for all UAVs allows us to extend the capability of the MPC tracker to avoid future collisions. When communication between the aircraft is established, they exchange their future trajectory predictions and act according to a decentralized mechanics, which will alter their courses to avoid the collision, based on sorting the UAVs by priorities. If there is a potential collision between two UAVs, the UAV with lower priority will avoid the other UAV by changing to a higher flight level. The system also allows priorities to be reassigned dynamically in the following cases:

- UAV should be avoided at all times (its priority is higher by definition). This may occur when it is currently grasping an object, or when its avoidance mechanism is accidentally turned off.

- UAV should avoid the other aircraft even if it has higher priority. Such a situation occurs when the other machine does not comply with the mechanics for any reason.

3.7 | High-level trajectory planning

High-level trajectory planning is built on top of MPC-based *Trajectory tracking*, which is used for precise tracking of the planned trajectories. The onboard online trajectory planning mechanism is used in two main parts of the Treasure Hunt scenario. The first task is Sweeping of the arena, where the team of UAVs is required to localize the objects within the arena, and either save their locations to the global map (at the beginning of the mission) or immediately try to grasp the first detected object (later in the mission, once all objects detected in the initial map have been processed, the grasping was successful, or failed repeatedly). The second online trajectory planning is utilized in Proactive collision-free planning, which is involved in cases where one UAV has to fly into another position. For example, when it holds the grasped object and wants to drop it into the dropping box.

3.7.1 | Sweeping

Sweeping the arena designed for the MBZIRC Treasure Hunt challenge involves localizing both dynamic and static objects. The trajectory planning for so-called sweeping can be described as coverage path planning (CPP; Galceran & Carreras, 2013), where for a given area the CPP should find a path from which the entire workplace can be scanned with an onboard sensor, in our case an onboard camera.

The proposed multirobot CPP algorithm is based on simple area decomposition into three equally large zones that split the area along the larger side (Figure 4). Each arena zone has one UAV assigned to localize and pick up the objects from. All UAVs then plan the coverage path using Boustrophedon coverage (Choset & Pignon, 1998) in each part of the area separately. Using Boustrophedon coverage, we create zigzag paths, as shown in Figure 5, such that the reduced FOV entirely covers the particular arena zone. The reduced FOV is set based on the required overlap in the coverage (set to 20% during the competition) and on the real FOV camera projection to the ground plane with respect to the sweeping altitude that is used.

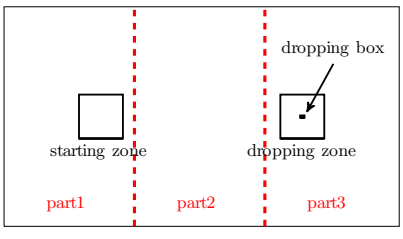


FIGURE 4 Decomposition of the Mohamed Bin Zayed International Robotics Challenge arena into three equally large zones [Color figure can be viewed at wileyonlinelibrary.com]

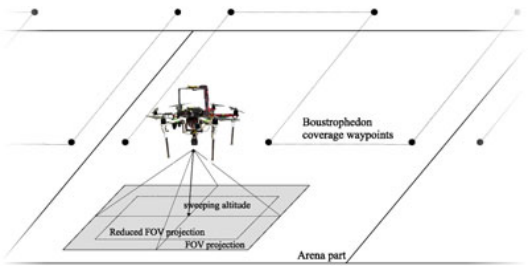


FIGURE 5 Boustrophedon coverage of the decomposed competition arena. FOV: field of view [Color figure can be viewed at wileyonlinelibrary.com]

To produce smooth trajectories for constant speed object detection, the Dubins vehicle model (Dubins, 1957) is used to create the final path between the waypoints. The minimal turning radius $\rho = v_c^2/a_{\max}$ of the Dubins vehicle was selected based on the desired constant velocity v_c (~ 3 ms^{-1}) and the maximal acceleration of the UAV $a_{\max}$ (~ 2 ms^{-2}), using an equation of circular motion with constant speed. The sweeping high-level trajectory planning is summarized in Figure 6, where the shown trajectories for all three UAVs were further used in the two following approaches in different stages of the Treasure Hunt scenario.

In the first approach, called static sweeping, the UAVs follow the created trajectories at a height (~ 7 m) and simultaneously detect the colored objects while the global map of the static objects is being created. After this initial coverage, the approximate positions of the detected static objects are estimated based on multiple detections of the same object. The second approach, called dynamic sweeping, is applied later in the schedule of the task, and the UAVs use similar paths as in the static sweeping. However, the sampled trajectories are used repeatedly (not just once, as in the static sweeping) and the UAVs do not create a global map. Instead, each UAV tries to find and estimate the position of any object while following the sweeping trajectory. When any object is located, the trajectory following is stopped and the UAV tries to grasp the object immediately. Either after successful grasping and dropping of the object, or after a number of unsuccessful

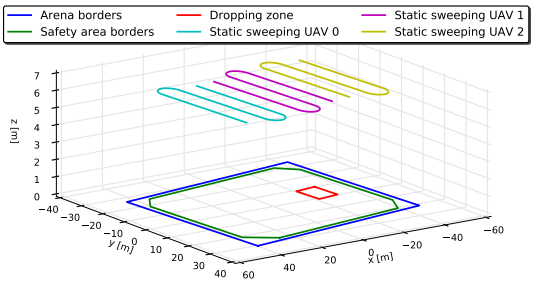


FIGURE 6 Sweeping trajectories based on Boustrophedon coverage using the Dubins vehicle and decomposition of the arena into three distinct parts, one for each UAV. UAV: unmanned aerial vehicle [Color figure can be viewed at wileyonlinelibrary.com]

grasps, the UAV continues with dynamic sweeping from the last trajectory sample.

3.7.2 | Proactive collision-free planning

Our strategy for covering the Treasure Hunt competition arena is based on decomposition into three equally large zones for each of the UAVs (Figure 4). Unfortunately, the dropping zone is located in one-third of the competition arena. After successful grasping, the UAV in part 1 therefore has to fly through the remaining zones to drop the object. Because there is a possibility of colliding with another UAV during this flight through the remaining zones, proactive collision-free planning has to be used. The actual positions of the UAVs are known due to information sharing, as was explained in Section 3.4. However, the Wi-Fi communication infrastructure is not reliable and, as mentioned, a multirobot system deployed in real world conditions should be robust to losing Wi-Fi communication. Therefore, we decided to use different flying heights for each of the UAVs, which minimizes the possibility of a collision, without any additional planning. Unfortunately, while completing this task the UAVs cannot maintain only these heights during the mission, as they have to descend for events such as grasping the objects and then dropping them. These events take most of the overall flight time, because they require a complicated grasping manoeuvre and hovering in front of the dropping zone, if it is sharing with other UAVs. Moreover, the grasping manoeuvre can be repeated several times before the object is gripped.

The proposed solution for finding a collision-free trajectory uses four assumptions derived from the MBZIRC rules, which are, however, valid for most cooperative transport applications:

- A Collision can occur only if a UAV leaves its dedicated height.
- The position of the UAV in the x-axis and in the y-axis does not alter rapidly in the event that it flies out of its safe altitude (the grasping and dropping manoeuvres are carried out following strictly vertical trajectories that accept grasping of dynamic objects, but where the lateral movement is also minor).
- The shape of the competition workspace is convex.
- At most three UAVs are used in the environment (this assumption is valid only for the MBZIRC Treasure Hunt task, but an extension of the approach is straightforward for different numbers of robots).

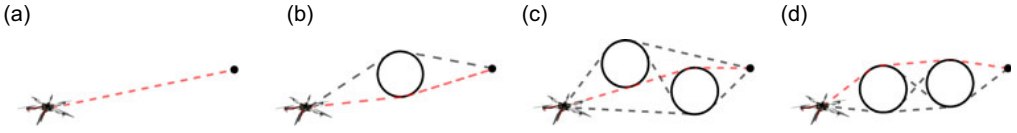
Thanks to these assumptions, the method for very rapidly computing a collision-free trajectory can be simplified to finding a collision-free path in 2D (at the dedicated height) between two points, where only two obstacles can occur. These obstacles are circles centered on the x and y coordinates of neighboring UAVs with safety radius r_a. It is prohibited to encroach on these circles. The safety radius of the circles depends on the speed of the UAVs which, for security reasons in the MBZIRC competition, was restricted to a maximum of 30 km/hr. We used a detection radius (the relative distance between UAVs in which the avoidance maneuver is initiated) of 5 m radius during the competition, while the critical radius in which the UAVs are considered to be in a collision is 0.8 m.

Based on the previously realized experimental comparison of available path planning approaches (Saska, Kulich, & Preucil, 2006), a visibility graph method (Lozano-Pérez & Wesley, 1979) was applied to solve the collision-free planning problem. The method provides the shortest path and it is sufficiently fast in simple situations including limited number of obstacles. Only four possible paths in the graph consisting of tangent lines to circles, which represent the obstacle, and the circle segments can be considered as a candidate solution in our case of two obstacles. The solution can, therefore, be found analytically in a very short time (possibly in each control step) with negligible burden on the processor. See examples of trajectories generated by proactive collision-free planning in Figure 7a–d.

A collision-free trajectory exists only for described planning when the start points or the end points are not inside the safety radius r_a of another UAV. In situations when a UAV is already inside the safety radius r_a of another UAV, the UAV finds a plan into the nearest position that is not in conflict with a UAV, and the collision-free planning procedure is initiated. If the high-level planning system requires to fly into a position, which is occupied by another UAV, then a temporary goal position is set instead. This position is the closest feasible position to the original goal such that it lies on the original trajectory. The UAV then waits for up to 1.5 min until the goal position is available again. If the goal position is not freed within this time, it is assumed that the information about the occupation of the goal position is incorrect. During the MBZIRC competition, the planning was repeated five times per second, and in the event of a communication interruption, the last received states of other UAVs were considered as correct for 5 s.

3.8 | Failure recovery and synchronization jobs manager

The main core of the system is the FSM concept, which is used for managing all subsystems. It increases the robustness of the entire



A path without obstacles. A path with one obstacle. A path with two obstacles. A path with two obstacles.

FIGURE 7 Examples of trajectories generated by fast proactive collision-free planning. The black circles denote obstacles. The red path shows the shortest collision-free trajectory, and the gray paths denote other collision-free paths [Color figure can be viewed at wileyonlinelibrary.com]

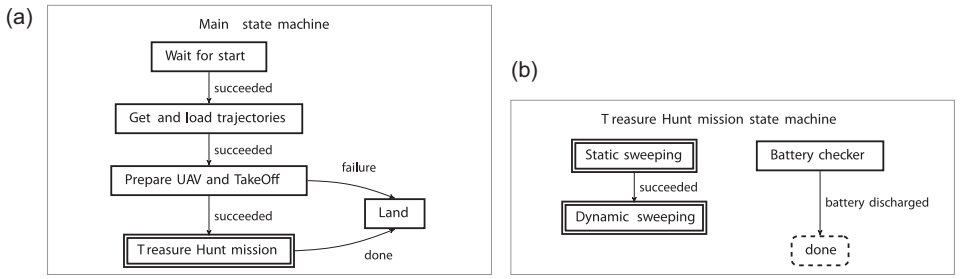


Diagram of the main state machine. Diagram of the Treasure Hunt mission state machine.

FIGURE 8 The structure of the FSM tool. FSM: failure recovery and synchronization jobs manager; UAV: unmanned aerial vehicle

code structure resolving the remaining few subsystem failure cases due to wrong sequential and concurrent operations. In the proposed system, the FSM is designed using SMACH (Bohren, 2017), a ROS-independent Python library, and it is fully integrated into the designed ROS framework.

As was mentioned in the introduction, the entire FSM structure may be considered as a hierarchical state machine with synchronization and failure recovery abilities. For simplicity, we will refer to the components of the FSM as state machines in this section. In Figures 8, 9, 10, and 12, the internal states of the FSM levels (the so-called state machines) are visualized by rectangles, and the nested lower-level state machines are visualized by double-line rectangles, such as the *Treasure Hunt mission* state machine introduced in Figure 8a by the *Treasure Hunt mission* rectangle, and described in detail in Figure 8b. Transitions between two states and from one state to a lower-level state machine are marked by the arrow with a label of an outcome describing the transition. Dotted terminal states represent the transition that is called after returning to a higher level state machine. The land event is called whenever any state produces an outcome that means that the UAV cannot continue in its mission.

The diagram of the main state machine is visualized in Figure 8a. In the first step, the trajectories for static sweeping and also for dynamic sweeping in the predefined part of the competition arena (see Section 3.7.1) are loaded, and an automatic take off is called. Once the UAV is

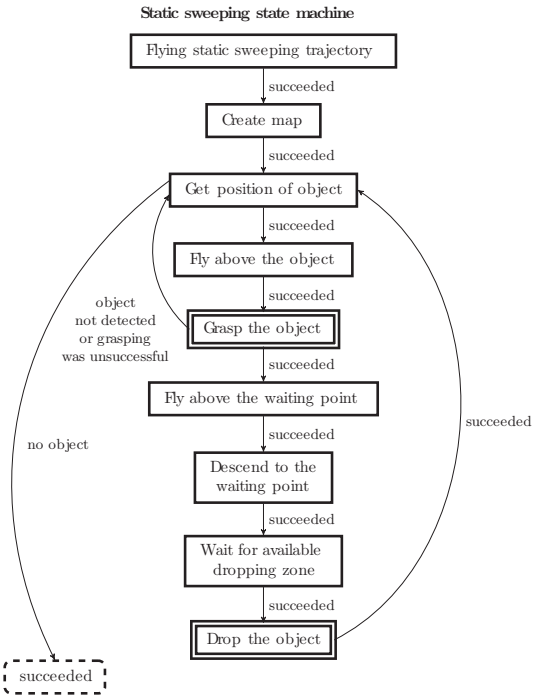


FIGURE 9 Diagram of the static sweeping state machine

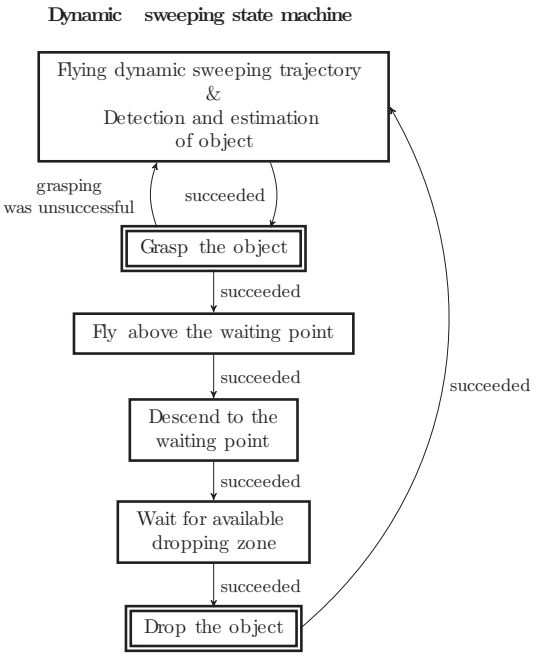
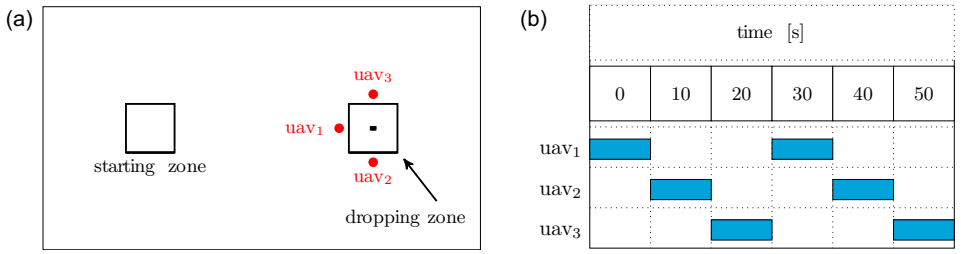


FIGURE 10 Diagram of the dynamic sweeping state machine



Position of the waiting point in the MBZIRC arena.

Gantt diagram of the time window for particular UAVs.

FIGURE 11 Waiting position around the dropping zone, and a Gantt diagram of the proposed time window strategy. The index of the UAV indicates to which part of the arena the UAV belongs. (a) Position of the waiting point in the MBZIRC arena. (b) Gantt diagram of the time window for particular UAVs. MBZIRC: Mohamed Bin Zayed International Robotics Challenge; UAV: unmanned aerial vehicle [Color figure can be viewed at wileyonlinelibrary.com]

in the air, the mission state machine is activated (Figure 8b). The mission is a concurrent state machine that sequentially runs the static sweeping procedure and the dynamic sweeping procedure, while simultaneously controlling the voltage of the battery. If the battery is discharged, the state machine terminates all currently executed tasks of the UAV, and a land event is called. The level of voltage for battery discharge was set experimentally for each battery type.

The static sweeping state machine (Figure 9) starts by following the sweeping trajectory and creating a map with approximate positions of the static objects. After this initial coverage of the competition arena, an attempt is made to grasp the nearest estimated static object in the map. The grasping procedure is shown in Figure 12. Initially, the state machine starts with the object detection mechanism. Whenever an object is located, the UAV tries to align itself horizontally above the estimated position of the object and then to descend to the grasping height of 1.5 m above the ground. Once the UAV has reached the desired height and it is aligned above the object, it tries to grasp the object. Whenever the object is lost in the steps after descending to the grasping height, the UAV ascends and repeats these steps again. The steps are also repeated if the grasping fails. Only two attempts are made to grasp the estimated object. If the UAV was not successful in these attempts, the state machine returns the UAV to the safe flying height and it is terminated with the outcome that the grasping was unsuccessful. After a successful grasp, the UAV also ascends to the safe flying height, but the grasping state machine outputs that the grasp was successful. The decision as to whether the UAV is carrying an object is made via a feedback from the Hall effect sensors that are placed on the gripper. To avoid deadlock, the state machine is terminated in the first node if the object is not found within a certain time.

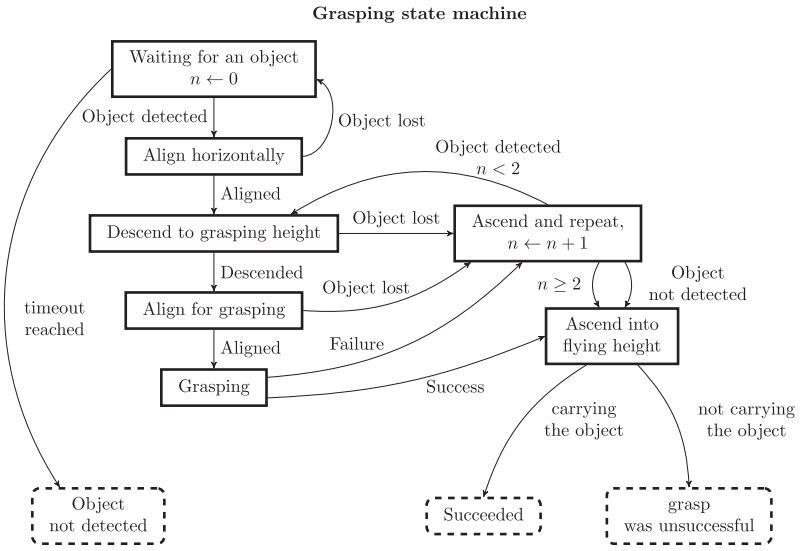


FIGURE 12 Diagram of the grasping state machine

The static sweeping state machine reacts to unsuccessful outcomes from the grasping procedure by selecting a new object for grasping from the map. When the grasping attempt was successful and the UAV is carrying the object, the UAV flies at its safe height to a position above the waiting point. The waiting point is selected based on the part of the competition arena assigned to the UAV. During the MBZIRC competition, the safe flying heights for part 1, part 2, and part 3 were 3, 4, and 5 m, respectively. The waiting points were located 7 m (measured in x, y plain) from the center of the dropping box (Figure 11a). Once the UAV reaches the position above the waiting point at its safe height, it descends to the dropping height of 1.5 m above the ground. At this waiting point, the UAV hovers until the moment when the dropping zone is not occupied by any other UAV, if the communication infrastructure is available, or until the UAV has access to the dropping zone based on the time windows, if the communication channel cannot be used for negotiation and for sharing the status of the dropping zone.

The negotiation about access to the dropping zone is based on queries of the current UAV on its waiting position addressed to neighboring UAVs. The neighboring UAV responds with confirmation that allows the current UAV to access the zone, but only when the neighboring UAV is not inside this zone, or if the neighboring UAV has not been waiting for access for longer than the current UAV. The current UAV starts with the dropping maneuver only when it receives confirmations from all neighboring UAVs. The negotiation about access to the dropping zone is carried out repeatedly until the UAV receives confirmations.

If communication has been lost for more than a predefined time during the mission, all UAVs will switch to a strategy with time windows for accessing the zone to avoid collisions in the dropping zone. Time windows 10 s in length are used for each UAV. This range of time windows provides two time intervals for dropping for each UAV per minute. The UAV in part 1 can be in the restricted area around the dropping zone in the 0–9 s time interval, the UAV in part 2 can be there in the 10–19 s time interval, and the UAV in the part 3 can be there in the 20–29 s time interval. This strategy is the same for accessing the dropping area in the second half of the minute, so the intervals are offset by 30 s (Figure 11b). The UAV can call the dropping procedure only when it is in the waiting position at the dropping height, and its time window starts. This strategy is not as effective as negotiation and sharing of the status of the dropping zone, but it is safer in the case of a problematic communication network. This strategy requires the clocks on the UAVs to be initially synchronized within a few milliseconds using chrony—an implementation of the network time protocol.

The dropping maneuver is done in sequence: Flying above the dropping box at the dropping height, dropping the object, and returning to the UAV safe height above the waiting position. After dropping the object, the state machine initializes the grasping procedure with the next estimated object in the map. This is done until all detected objects have been grasped, or an attempt has been made to grasp them, in the case of a grasping failure.

In the dynamic sweeping state machine (Figure 10), the UAV flies the dynamic sweeping trajectory, and when any object is detected and its position is estimated, the UAV immediately tries to grasp it. After successful grasping and dropping, the UAV flies back into the dynamic sweeping trajectory and continues with dynamic sweeping while simultaneously looking for the remaining objects. This approach is not as effective as the initial static sweeping procedure, where the UAVs could fly for another object in the map directly, and minimize the overall flight time, but it is more robust. In the ideal case of perfect mapping and grasping procedures, all static objects are grasped during the static sweeping part, and only the dynamic objects are hunted during the dynamic sweeping. In the demanding real-world conditions of the MBZIRC arena, with changing light conditions and wind gusts, many objects were not grasped in the first phase of the mission. This was due to a safety procedure that allowed a limited number of grasping attempts per object to avoid a deadlock. These missed objects could be grasped later, in the dynamic sweeping part, as the local environment conditions changed.

Another interesting property of this approach is the possibility to exchange the sweeping trajectories, and therefore the operational zones and waiting positions between the UAVs after a given period of the mission. This increases the robustness and the performance of the overall system in the event of a failure or a malfunction of a UAV subsystem. Even if all components of all UAVs are fully functional as designed, each UAV in the team behaves differently in different tasks, and it often happened that a UAV could accomplish a task in which another team member failed, and vice versa. This is another useful lesson learned during the MBZIRC event that should be adapted for designing multirobot systems, if possible. Finally, splitting the static object grasping in the initial sweeping part and the subsequent grasping of dynamic objects and the remaining static objects increases the overall system robustness. There is a much lower probability of a UAV crash during static object grasping. This has been shown in numerous realistic complex simulations, and also during system testing and its deployment in the competition.

4 | EXPERIMENTAL RESULTS

In this section, we present both the experimental results achieved while preparing for the Treasure Hunt scenario, and also the performance of the system during the MBZIRC competition. The remainder of this section is divided into main parts, where we present the experimental results achieved in the simulator, during the preparations for the competition in South Bohemia, in the final tests in a challenging desert environment and in the course of the MBZIRC competition. A video attachment to this paper is available at website http://mrs.felk.cvut.cz/jfr2018treasurehunt.

4.1 | Robotic simulator

The system was initially developed using the Gazebo robotic simulator, which was used as the simulation in the loop, together

TABLE 1 Percentiles of duration of the experiment before the first collision occurred

Percentile	0.5	0.75	0.95	0.99
Without the avoidance	104 s	152 s	264 s	431 s
With the avoidance	–	–	–	–

Note. The results were obtained in two 24-hr simulated flights (one with and one without the collision avoidance mechanism used) with five UAVs, conducting a 2D random walk on 100 × 100 m area. The total of 495 collisions were recorded if the collision avoidance mechanism was not used.

with the PixHawk firmware. Using the robotic simulator, the process of developing the subsystems and integrating the entire system was carried out significantly faster and more safely than when using the real system directly. In addition, by modeling the whole scenario in the simulator and by testing the behavior of the complex FSM approach in it, the complete system achieved the necessary level of reliability for deployment in tasks such as the Treasure Hunt.

The underlying layers of the control pipeline, namely, the UAV state estimation, control, tracking, and predictive collision avoidance, were extensively tested using the Gazebo simulator. To show the system robustness, we conducted 24-hr simulated flights of five UAVs in an area of 100 × 100 m. Each UAV followed an independent random walk reference in the same height. Without the collision avoidance technique, the median time of the first collision between any of the UAVs was 104 s, from total of 495 simulated scenarios (simulation was always restarted after the first collision). With the collision avoidance mechanism, there was not a single collision within the 24 hr of the experiment, while the minimal registered distance between the UAVs was 1.21 m, which is still 50% more than the collision distance 0.8 m of the used platforms. See Table 1 for the comparison of percentiles of duration of the experiment before the first collision occurred.

The results from 20 simulations of the complete MBZIRC 2017 Treasure Hunt scenario are shown in Table 2. Each of these simulations contained 10 static and 10 dynamic objects, which were randomly placed in a simulated MBZIRC arena. Snapshots from simulation are shown in Figure 13. We expected that the dynamic objects will move according to some motion model that is predictable

TABLE 2 Results from 20 simulations of Challenge 3, in which the objects (10 static and 10 moving) were randomly placed

	Mission time (min)	Time needed for grasping of the static object (s)	Time needed for grasping of the dynamic object (s)	Smallest distance between UAVs (m)
Min	12.1	23.7	35.0	1.9
Max	17.4	36.4	51.2	3.3
Mean	13.6	30.6	43.6	2.5

Note. UAVs in a distance closer than 0.8 m are colliding in the simulation as well as in the real system, which never happened in simulations and real flights if the collision avoidance approach was used.
UAV: unmanned aerial vehicle.

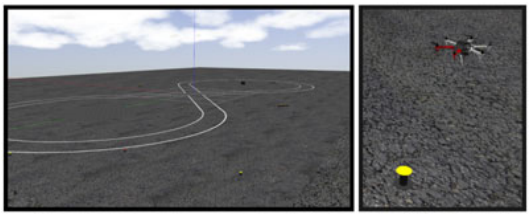


FIGURE 13 Snapshots from the simulation developed for the Mohamed Bin Zayed International Robotics Challenge competition [Color figure can be viewed at wileyonlinelibrary.com]

and smooth. Therefore, we modeled the movement of the dynamic objects in simulation using the car-like motion model, where the velocity of the object did not exceed 0.3 m/s. Due to the movement type of dynamic objects being uncertain, the mission was divided into two parts. The first part is the safe part of the mission, where only the static objects are attempted to be grasped and delivered. After this part is finished, the rest of objects will be targeted regardless of whether they are static or dynamic. Results from the simulations in Table 2 show that the system is capable of collecting all targets to the dropping box in the competition time interval of 20 min. The best time of finishing the mission was 12.1 min and the worst was 17.4 min. The average time needed was 13.6 min. Results in Table 2 also show that all static objects were grasped faster than the fastest dynamic object. Furthermore, thanks to using collision avoidance methods, there was no collision between members of the team during the mission. The closest any UAVs got to each other was 1.9 m, which only happened in one of the simulations, and in general the mutual distances were higher than that.

4.2 | Experimental camps in the Czech Republic

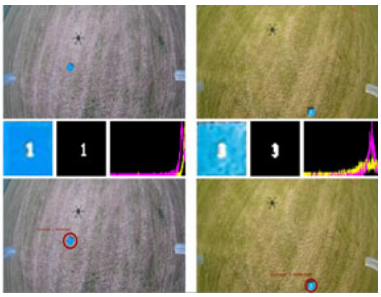
Key parts of the proposed system were tested in the course of experimental camps held in the countryside of South Bohemia in the Czech Republic throughout the year before the competition (Figure 14). Repeated experimental verification of key parts of the proposed system was necessary to test phenomena that are difficult to simulate, and also to discover issues that were not present in our previous hardware experiments without physical interaction of the robot with the real-world environment. One issue that was discovered was the influence of the force produced by the propellers on the carried objects. This exposed the need for a stronger magnetic gripper, which we then designed. Another discovered issue was the ground effect caused by the objects. This manifested itself as turbulence in the last phase of the grasping maneuver.

The most crucial parts of the system were the low-level UAV control and the MPC-based trajectory tracking, used for precise positioning of the UAV. These were thoroughly tested to obtain the centimeter precision required for the grasping task. The MPC-based trajectory tracking used during the colored object mapping is shown in Figure 14a. In addition, initial testing of the object detection was carried out. However, in accordance with the initial specification of the

(a)

MPC-based trajectory tracking with low-level UAV control during the mapping of the colored objects spread throughout the experimental field.

(b)



Object detection and number (reward) recognition of square-shaped colored objects.

FIGURE 14 Experimental verification of the MPC-based trajectory tracking method and the object detection algorithm during the experimental camps in the countryside of South Bohemia in the Czech Republic. http://mrs.felk. cvut.cz/jfr2018treasurehunt-video1. MPC: model predictive control; UAV: unmanned aerial vehicle [Color figure can be viewed at wileyonlinelibrary.com]

shape of the object expected in the competition, we designed square shaped objects with numbers describing the rewards (Figure 14b). Videos showing initial attempts for grasping and dropping of the object, and trajectory following are available at http://mrs.felk.cvut.cz/jfr2018treasurehunt-video1.

One of the experimentally verified subsystems was the MPC-based collision avoidance implemented for reactive avoidance of collisions between multiple UAVs in the MBZIRC competition. Using the MPC predictions of the future parts of the trajectory of other UAVs (discussed in Section 3.6), each UAV can avoid collisions with other UAVs by a simple change of flight height in potential collision parts of the future trajectory. It is necessary to employ this method in scenarios with a problematic communication network. This is because after reestablishing communication the proactive collision-free planning may not be able to deal with a suddenly discovered imminent collision, or may not even be active in the current phase. This safety mechanism is implemented on the lowest level of control in all phases of the mission. Figure 15 shows the verification of MPC-based collision avoidance, with two UAVs exchanging their position and one hovering UAV between the two positions. A video showing this verification is available at http://mrs.felk.cvut.cz/jfr2018treasurehunt-video2. Such collision avoidance requires only a small number of messages to be shared between UAVs. These messages contain the MPC future trajectory predictions of each UAV, and are distributed with a very low frequency of 2 Hz. Although the proposed collision avoidance technique requires only a low communication bandwidth (~ 6 kB/s for three UAVs), the collision avoidance was not always used during the competition, due to dropouts of communication between UAVs, which was observed by all teams in the competition.

Another evaluated subsystem was the object detection and mapping. In particular, the datasets gathered were used to compare computational efficiency of our object detection method to the Maximally Stable Extremal Regions (MSER) (Matas, Chum, Urban, & Pajdla, 2004) and "SimpleBlobDetector" methods included in the OpenCV library (Bradski, 2000). The results indicated that the system presented achieved significantly higher frame rates compared to the aforementioned two methods. This confirmed the experiments in Krajník et al. (2014), which introduced an algorithm our detection was based on.

4.3 | Desert testing in the United Arab Emirates

Finally, the complete system was thoroughly tested for a period of three weeks just before the competition, in the desert near Abu Dhabi in the United Arab Emirates. The desert environment was challenging, due to the uneven terrain and the rapidly changing wind conditions. By tuning the system for such weather and terrain conditions, our system was better prepared for the environment at the Yas Marina Circuit in Abu Dhabi, where the competition was held. The rapidly changing terrain profile in the dunes of the desert also had an influence on the quality of the communication network. The frequent interruptions of the connection inspired our solution, which does not rely on the communication network.

As we have mentioned, several important features of our system were, in our opinion, the dominant factors that led to our winning performance in all trials of the Treasure Hunt challenge in the MBZIRC competition. Most of the other teams did not take into consideration external disturbances such as wind in their controller. Surprisingly, the MBZIRC competition arena was not perfectly flat, and some teams had relied on its flatness. Finally, relying on a robust communication network was the main bottleneck of the competitive solutions.

Photos from the tests of the system in the desert are shown in Figure 16. The grasping procedure is captured in the image on the right, and the dropping maneuver is shown in the image on the left. A video showing the behavior of the complete system with three UAVs in this environment is available at http://mrs.felk.cvut.cz/jfr2018treasurehunt-video3. During this testing, the yellow objects were stationary as opposed to the competition, where they were dynamic. This means, that in this phase, the system was tested for the static

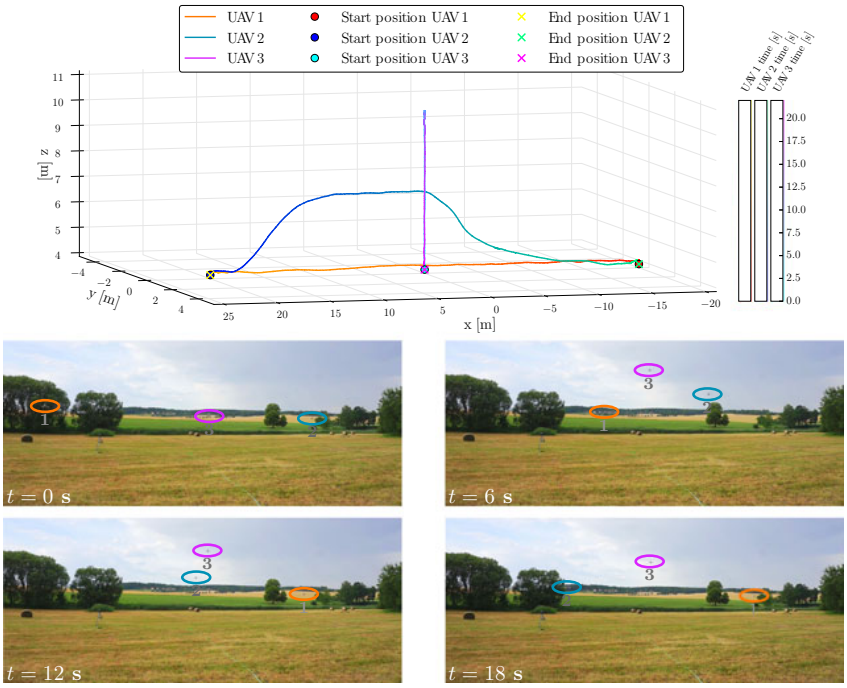


FIGURE 15 MPC-based collision avoidance between three drones. Two drones (UAV 1 and UAV 2) exchange their positions, while the third UAV 3 hovers in a position colliding with their trajectories. Using MPC future trajectory prediction, the UAVs avoid a collision by changing their trajectory height. http://mrs.felk.cvut.cz/jfr2018treasurehunt-video2. MPC: model predictive control; UAV: unmanned aerial vehicle [Color figure can be viewed at wileyonlinelibrary.com]

objects only. In addition, the paths traveled by the UAVs during the experiment presented in the video are shown in Figure 17. In this figure, the z-axis denotes the height above the level of the starting position as measured by the differential RTK GPS. The UAVs were kept at constant height above the ground and therefore the graph shows how uneven the terrain was. Furthermore, Figure 17 depicts the positions and colors of the objects that were collected.

4.4 | Results from the MBZIRC competition

Our system was applied four times in the Treasure Hunt scenario during the final MBZIRC competition. During the competition, the number of dynamic (yellow) objects was decreased from announced 10 to 3 for this scenario for organizational reasons. The results, that is, the number of colored objects that were collected, are shown in Table 3. The first two

FIGURE 16 Photos from the tests of the proposed system in the desert near Abu Dhabi, United Arab Emirates. The grasping procedure is captured in the image on the right, and the dropping maneuver is shown in the image on the left. http://mrs.felk.cvut.cz/jfr2018treasurehunt-video3 [Color figure can be viewed at wileyonlinelibrary.com]

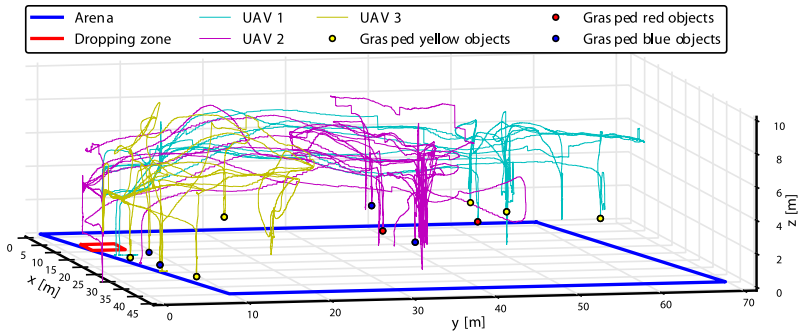


FIGURE 17 The paths traveled by individual UAVs during the desert experiment. The colored points denotes the positions of the objects that were collected. UAV: unmanned aerial vehicle [Color figure can be viewed at wileyonlinelibrary.com]

TABLE 3 Numbers of the objects collected in the Treasure Hunt scenario during the MBZIRC 2017 competition

	Placed into the box	Placed outside the box but inside the dropping area
TRIAL 1	2R, 2G	1G
TRIAL 2	2R, 3G	
GRAND 1	1R, 1G, 2B, 1Y	1G, 1B
GRAND 2	2R, 3G, 1B, 1Y	1G

Note. B: blue static object; G: green static object; GRAND 1 and 2: trials of the MBZIRC Grand Challenge; R: red static object; TRIAL 1 and 2: trials of MBZIRC Challenge 3; Y: yellow nonstationary object.

attempts, denoted as TRIAL 1 and TRIAL 2, are the results from Challenge 3, which contained only the Treasure Hunt scenario. The remaining two trials (GRAND 1 and GRAND 2) were a part of the Grand Challenge, where the Treasure Hunt scenario was undertaken simultaneously with the scenario of landing on a moving ground vehicle (Challenge 1), and the scenario where a ground robot had to locate and reach a panel, and further physically operate a valve located on the panel (Challenge 2). During these four trials within the competition, 25 objects overall were successfully placed into the dropping zone. The best performance according to the number of grasped and placed objects was achieved during the second trial of the Grand Challenge, when eight objects, including a nonstationary object were brought into the dropping zone. The system described in this paper won first place in Challenge 3, and contributed to our third place in the Grand Challenge. A video showing results from the MBZIRC competition is available at http://mrs.felk.cvut.cz/jfr2018treasurehunt-video4.

One part of the system for the Treasure Hunt scenario involved localizing objects using sweeping trajectories (described in Section 3.7.1). The static sweeping paths traveled by UAVs in the trials of Challenge 3 are shown in Figures 18 and 19. The flight time of the described UAV platform with fully charged four cell batteries with 6,750 mAh capacity is up to 15 min, which is less than allowed time per trial. The organizers allowed to change the batteries during the trial without any penalization. The trajectories before changing the batteries are labeled in the graphs as part 1, and after the batteries are changed, they are labeled as part 2. Furthermore, on these graphs, the colored points denote the detections of the objects that were observed, and the larger circles denote the estimated positions of these objects. After processing the data from the first trial, we decided to decrease the sweeping trajectory height from 7 to 6.5 m. This modification made objects more visible in camera images, which improved object detection. A disadvantage of this change

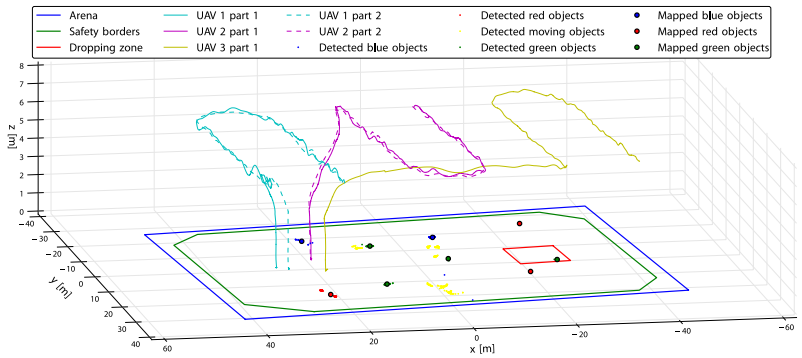


FIGURE 18 Mapping sweep during the first trial of Challenge 3. The colored points denote the detections of the objects that were observed, and the larger circles denote the estimated positions of these objects. UAV: unmanned aerial vehicle [Color figure can be viewed at wileyonlinelibrary.com]

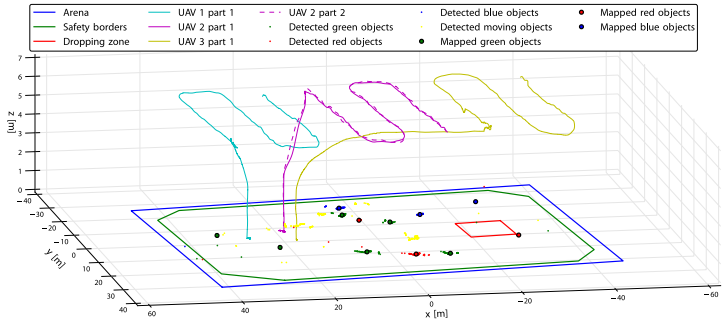
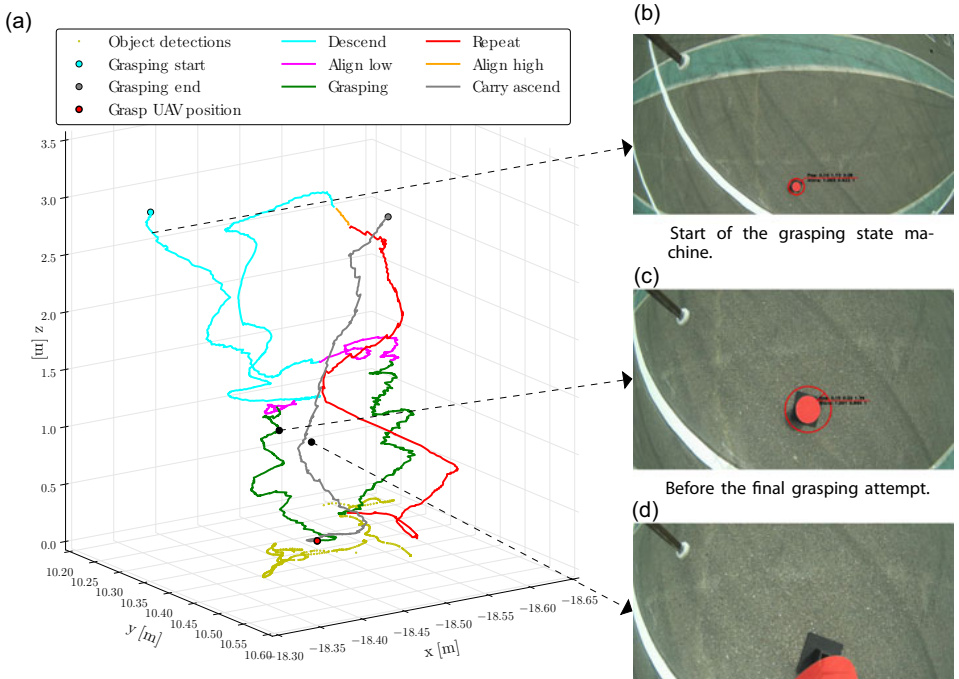


FIGURE 19 Mapping sweep during the second trial of Challenge 3. The colored points denote the detections of the objects that were observed, and the larger circles denote the estimated positions of these objects. UAV: unmanned aerial vehicle [Color figure can be viewed at wileyonlinelibrary.com]

was that it prolonged the trajectories, because the condition of at least 20% of overlap in the coverage could not be satisfied by following the same trajectory (in the *xy* plane). For this reason, the sweeping trajectories differ between these two trials.

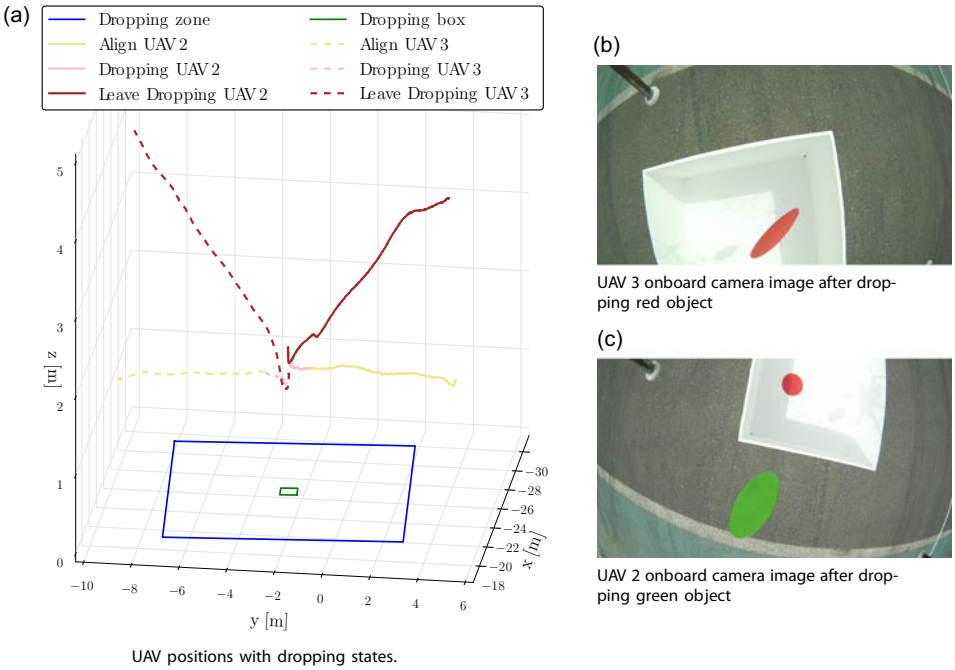
Another important part of described system is the grasping capability, where the UAV has to grasp a ferrous object. The overall grasping approach has been presented in Section 3.8, where the grasping state machine is depicted in Figure 12. Switching of the phases of the grasping state machine is shown in Figure 20, where an attempt at grasping was repeated after being aborted once. For a visualization of the transition between these phases, the resolution of the graph in Fig. 20a is 0.05 m in the *x*-axis and in the *y*-axis. In addition, detections of the object in three parts, which are indicated by dotted arrows, are shown in Figure 20b–d.

The dropping approach for delivering the grasped objects into the dropping box has been described in Section 3.8. Switching the phases of the dropping state machine is shown in Figure 21a, where the dropping procedure was carried out by two UAVs. Objects were dropped by each UAV at a different time, and thus there was no collision between them. Figure 21b,c show snapshots from the onboard cameras on the UAVs during the dropping maneuver.



UAV position with grasping states and object detections.

FIGURE 20 Phases of the grasping state machine, UAV position estimation and object detection during grasping of a red static object, with a successful second grasping attempt. UAV: unmanned aerial vehicle [Color figure can be viewed at wileyonlinelibrary.com]



UAV positions with dropping states.

FIGURE 21 Phases of the dropping state machine of two UAVs. UAV: unmanned aerial vehicle [Color figure can be viewed at wileyonlinelibrary.com]

Photos from the competition are shown in Figure 24. The upper image shows the UAV following the static sweeping trajectory. The images in the middle row and the image on the left in the lower part of the figure capture moments when the UAVs were grasping objects. The remaining image shows an object being dropped into the dropping box. In addition, the paths traveled by the UAVs during the first trial of Challenge 3 are shown in Figure 22, and the paths traveled in the second trial of the same challenge are shown in Figure 23. Furthermore, in these graphs, the colored points denote the positions of the objects that were collected.

5 | LESSONS LEARNED

Although the competition results can be considered a major success, it was not without hurdles, mainly during implementation, testing and tuning of the proposed system. From the implementation part, it was convenient to develop the system compatible with the ROS, which allows to divide the system into independent components that were implemented separately by different research groups. Furthermore, their testing were significantly simplified by employing the Gazebo robotic simulator together

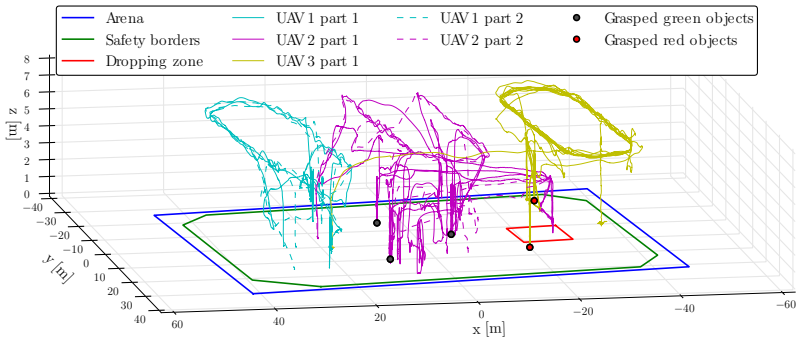


FIGURE 22 Paths traveled by individual UAVs during the first trial of Challenge 3. The colored points denotes the positions of the objects that were collected. UAV: unmanned aerial vehicle [Color figure can be viewed at wileyonlinelibrary.com]

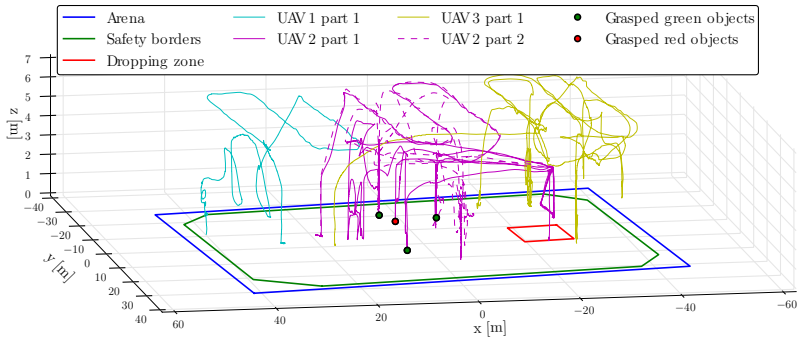


FIGURE 23 Paths traveled by individual UAVs during the second trial of Challenge 3. The colored points denotes the positions of the objects that were collected. UAV: unmanned aerial vehicle [Color figure can be viewed at wileyonlinelibrary.com]

with the firmware from PixHawk, which speeds up the overall progress of the development.

The required usage of more vehicles simultaneously even increased the complexity of the task. Every UAV is equipped with several sensors that could be a source of unreliability. By testing the behavior of the proposed system in desert near Abu Dhabi in the United Arab Emirates, our system was well prepared for the environment at the Yas Marina Circuit in Abu Dhabi, where the competition was held. The system was tuned to properly react to strong wind, decreased visibility due to sand, and to problems occurring by intensive light from sun. Hence we stress the significance of the real-world outdoor experiments above simulation, to obtain real sensor data.

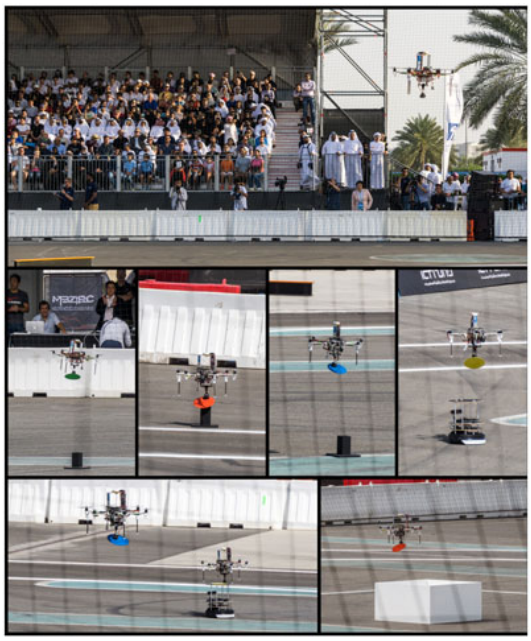


FIGURE 24 Photos from the MBZIRC competition. The upper image shows the UAV while following the static sweeping trajectory. The images in the middle row and the left on the lower part of the figure capture the moments when the UAVs were grasping objects. The remaining image shows an object being dropped into the dropping box. During four trials within the competition (two for Challenge 3 and two for the Grand Challenge), 25 objects overall were placed into the dropping zone (Table 3). http://mrs.felk.cvut.cz/jfr2018treasurehunt-video4. MBZIRC: Mohamed Bin Zayed International Robotics Challenge; UAV: unmanned aerial vehicle [Color figure can be viewed at wileyonlinelibrary.com]

5.1 | Toward a more general solution

Despite our best effort to develop a general solution capable of autonomous searching, picking, and placing objects, several subsystems have been tailored specifically to the competition scenario. The vision system was designed to locate the objects with colors and shapes specified by the rules of the competition. In the case of an object of more difficult shape and color patterns, a different approach for its localization would be required, for example, based

FIGURE 25 Team members that were involved in the MBZIRC competition in Abu Dhabi, United Arab Emirates. MBZIRC: Mohamed Bin Zayed International Robotics Challenge [Color figure can be viewed at wileyonlinelibrary.com]

on artificial neural networks. Further, estimation and prediction of the object movement using a car-like model provide a framework suitable for tracking the most common ground vehicles. A more precise model could be used to better estimate state of a specific vehicle (e.g. with differential drive model, or if capable of 3D motion). The presented proactive collision-free planning, using different flying heights and the visibility graph method, has been selected due to simple scenario with three UAVs. A requirement of a higher number of independent flying heights would occur with higher number of deployed UAVs. Then, a different splitting of the arena would be required since it is not efficient to often ascend and descend for the UAV. In this case, each individual UAV will be resolving a possible collision only with other UAVs, that are assigned to arena parts through which the UAV will need to fly. Taking these observations, the presented approach can be applied to various outdoor multirobot scenarios, as shown in our consequent research after the competition listed in Section 1.2.

6 | CONCLUSIONS

A system designed for Challenge 3 of the MBZIRC competition has been described in this paper. The paper has focused on the properties of the design that in our opinion were the most important factors leading to the best performance of the system in all trials in the 7 Treasure Hunt challenge. The system is able to solve object manipulation tasks in demanding outdoor environments, and to do so cooperatively in a team of three UAVs.

While many of the methods described here do not represent the bleeding edge of robotics research, they were designed to be versatile and substitutable. This allowed their easy integration into a complex modular system, which enabled efficient testing of the individual modules, making us aware of these modules deficiencies and possible faults during their deployment in real conditions. Our knowledge of the faults encountered during the field tests was reflected in the design of the core module of our system, the FSM. This module ensured that occasional faulty behaviour of the individual modules did not result in a critical failure or system deadlock. Still, the development of this complex system led to numerous significant contributions beyond the state-of-the-art in robotics, which could facilitate the deployment of multi-UAV platforms in challenging scenarios motivated by current needs of the industry. This was the main motivation for our paper and also for the MBZIRC competition itself.

The results shown in numerous realistic simulations in Gazebo and in experiments in a demanding desert environment have been presented in this paper following by analysis of necessary improvements of the system towards more general applications, which go beyond the MBZIRC 2017 competition. However, the most meaningful and credible verification of the performance and the reliability of the system was achieved in the MBZIRC competition, where our approach won the first place Challenge 3, on the basis of achieving the best score among 17 finalists selected from 142 registered teams.

ACKNOWLEDGMENTS

The outstanding results of this project could not have been achieved without the full cooperation of each member of our team, comprising people from the Czech Technical University in Prague, the University of Pennsylvania, and the University of Lincoln, UK (Figure 25). The work has been supported by CTU grant no. SGS17/187/OHK3/3T/13, Research Center for Informatics project no. CZ.02.1.01/0.0/0.0/16_019/0000765, the Grant Agency of the Czech Republic under grant no. 17-16900Y, ARL grant W911NF-17-2-0181, ONR grants N00014-07-1-0829, N00014-14-1-0510, and by Khalifa University.

ORCID

Vojtěch Spurný http://orcid.org/0000-0002-9019-1634
Tomáš Báča http://orcid.org/0000-0001-9649-8277
Robert Pěnička http://orcid.org/0000-0001-8549-4932
Tomáš Krajník http://orcid.org/0000-0002-4408-7916
Justin Thomas http://orcid.org/0000-0002-4253-417X
Dinesh Thakur http://orcid.org/0000-0001-5046-8160
Giuseppe Loianno http://orcid.org/0000-0002-3263-5401

REFERENCES

Afolabi, D., Man, K. L., Liang, H.-N., Guan, S.-U., & Krilavičius, T. (2015). 1543. monocular line tracking for the reduction of vibration induced during image acquisition. *Journal of Vibroengineering*, *17*, 655–661.

Baca, T., Loianno, G., & Saska, M. (2016, August). Embedded model predictive control of unmanned micro aerial vehicles. 21st International Conference on Methods and Models in Automation and Robotics (MMAR), 992–997. Miedzyzdroje, Poland.

Bähnemann, R., Schindler, D., Kamel, M., Siegwart, R., & Nieto, J. (2017). A decentralized multi-agent unmanned aerial system to search, pick up, and relocate objects. Proceedings of 2017 IEEE International Symposium on Safety, Security and Rescue Robotics (SSRR), 123–128. Shanghai, China.

Bohren, J. (2017). SMACH a task-level python execution framework for rapidly composing complex robot behaviors. Robot Operating System (ROS). etrieved from http://wiki.ros.org/smach. (cited on July 17, 2018).

Bradski, G. (2000). The OpenCV library. *Dr. Dobb's Journal: Software Tools for the Professional Programmer*, *25*(11), 120–123.

Choset, H., & Pignon, P. (1998). *Coverage path planning: The Boustrophedon cellular decomposition* (203–209). London: Springer London.

Digi International, I, (2017). XBee-PRO RF Modules. https://www.sparkfun.com/datasheets/Wireless/Zigbee/XBee-Datasheet.pdf. (cited on July 17, 2018).

Dubins, L. E. (1957). On curves of minimal length with a constraint on average curvature, and with prescribed initial and terminal positions and tangents. *American Journal of Mathematics*, *79*(3), 497–516.

Fumagalli, M., Stramigioli, S., & Carloni, R. (2016). Mechatronic design of a robotic manipulator for unmanned aerial vehicles. In 2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 4843–4848. Daejeon, South Korea.

Galceran, E., & Carreras, M. (2013). A survey on coverage path planning for robotics. *Robotics and Autonomous Systems*, *61*(12), 1258–1276.

Ghadiok, V., Goldin, J., & Ren, W. (2012). On the design and development of attitude stabilization, vision-based navigation, and aerial gripping for a low-cost quadrotor. *Autonomous Robots*, *33*(1), 41–68.

Gioioso, G., Franchi, A., Salvietti, G., Scheggi, S., & Prattichizzo, D. (2014). The flying hand: A formation of uavs for cooperative aerial

tele-manipulation.2014 IEEE International Conference on Robotics and Automation (ICRA), 4335–4341. Hong Kong, China.

Heredia, G., Jimenez-Cano, A. E., Sanchez, I., Llorente, D., Vega, V., Braga, J., … Ollero, A. (2014). Control of a multirotor outdoor aerial manipulator. In 2014 IEEE/RSJ International Conference on Intelligent Robots and Systems, 3417–3422. Chicago, IL.

Herissé, B., Hamel, T., Mahony, R., & Russotto, F. X. (2012). Landing a vtol unmanned aerial vehicle on a moving platform using optical flow. *IEEE Transactions on Robotics*, *28*(1), 77–89.

Kamel, M., Comari, S., & Siegwart, R. (2016). Full-body multi-objective controller for aerial manipulation. In 2016 24th Mediterranean Conference on Control and Automation (MED), 659–664. Athens, Greece.

Kannan, S., Quintanar-Guzman, S., Dentler, J., Olivares-Mendez, M. A., & Voos, H. (2016). Control of aerial manipulation vehicle in operational space. In 2016 8th International Conference on Electronics, Computers and Artificial Intelligence (ECAI), 1–4. Ploiesti, Romania.

Kessens, C. C., Thomas, J., Desai, J. P., & Kumar, V. (2016). Versatile aerial grasping using self-sealing suction. In 2016 IEEE International Conference on Robotics and Automation (ICRA), 3249–3254. Stockholm, Sweden.

Kim, S., Seo, H., Choi, S., & Kim, H. J. (2016). Vision-guided aerial manipulation using a multirotor with a robotic arm. *IEEE/ASME Transactions on Mechatronics*, *21*(4), 1912–1923.

Korpela, C. M., Danko, T. W., & Oh, P. Y. (2012). Mm-uav: Mobile manipulating unmanned aerial vehicle. *Journal of Intelligent and Robotic Systems*, *65*(1), 93–101.

Krajník, T., Nitsche, M., Faigl, J., Vaněk, P., Saska, M., Přeučil, L., & Mejail, M. (2014). A practical multirobot localization system. *Journal of Intelligent and Robotic Systems*, *76*(3-4), 539–562.

Lee, T., Leok, M., & McClamroch, N. H. (2013). Nonlinear robust tracking control of a quadrotor uav on se(3). *Asian Journal of Control*, *15*(2), 391–408.

Lippiello, V., Cacace, J., Santamaria-Navarro, A., Andrade-Cetto, J., Trujillo, M. ., Esteves, Y. R., & Viguria, A. (2016). Hybrid visual servoing with hierarchical task composition for aerial manipulation. *IEEE Robotics and Automation Letters*, *1*(1), 259–266.

Lozano-Pérez, T., & Wesley, M. A. (1979). An algorithm for planning collision-free paths among polyhedral obstacles. *Communications of the ACM*, *22*(10), 560–570.

Matas, J., Chum, O., Urban, M., & Pajdla, T. (2004). Robust wide-baseline stereo from maximally stable extremal regions. *Image and Vision Computing*, *22*(10), 761–767.

Meier, L., Tanskanen, P., Heng, L., Lee, G. H., Fraundorfer, F., & Pollefeys, M. (2012). Pixhawk: A micro aerial vehicle design for autonomous flight using onboard computer vision. *Autonomous Robots*, *33*(1), 21–39.

Mellinger, D., & Kumar, V. (2011). Minimum snap trajectory generation and control for quadrotors. In 2011 IEEE International Conference on Robotics and Automation (ICRA), 2520–2525. Shanghai, China.

Mellinger, D., Lindsey, Q., Shomin, M., & Kumar, V. (2011). Design, modeling, estimation and control for aerial grasping and manipulation. In 2011 IEEE/RSJ International Conference on Intelligent Robots and Systems, 2668–2673. San Francisco, CA.

Mellinger, D., Shomin, M., Michael, N., & Kumar, V. (2013). *Cooperative grasping and transport using multiple quadrotors* (545–558). Berlin, Heidelberg: Springer.

Mobius ActionCam (2018). http://www.mobius-actioncam.com/mobius/wp-content/uploads/2015/01/Mobius-Manual-23jan15a.pdf. (cited on September 20, 2018).

Morton, K., & Toro, L. F. G. (2016). Development of a robust framework for an outdoor mobile manipulation uav.2016 IEEE Aerospace Conference, 1–8. Big Sky, MT.

NicaDrone (2017). OpenGrab EPM v3 electropermanent magnet. http://nicadrone.com. (cited on July 17, 2018).

Nieuwenhuisen, M., Beul, M., Rosu, R. A., Quenzel, J., Pavlichenko, D., Houben, S., & Behnke, S. (2017). Collaborative object picking and delivery with a team of micro aerial vehicles at mbzirc. In 2017 European Conference on Mobile Robotics (ECMR), 1–6. Paris, France.

Orsag, M., Korpela, C., Pekala, M., & Oh, P. (2013). Stability control in aerial manipulation. In 2013 American Control Conference, 5581–5586. Washington, DC.

Parra-Vega, V., Sanchez, A., Izaguirre, C., Garcia, O., & Ruiz-Sanchez, F. (2013). Toward aerial grasping and manipulation with multiple uavs. *Journal of Intelligent and Robotic Systems*, *70*(1), 575–593.

Pěnička, R., Faigl, J., Váňa, P., & Saska, M. (2017). Dubinsorienteering problem. *IEEE Robotics and Automation Letters*, *2*(2), 1210–1217.

Pěnička, R., Saska, M., Reymann, C., & Lacroix, S. (2017). Reactive dubins traveling salesman problem for replanning of information gathering by uavs. In European Conference of Mobile Robotics (ECMR), 1–6. Paris, France.

Pounds, P. E. I., Bersak, D. R., & Dollar, A. M. (2011a). Grasping from the air: Hovering capture and load stability. In 2011 IEEE International Conference on Robotics and Automation, 2491–2498. Shanghai, China.

Pounds, P. E. I., Bersak, D. R., & Dollar, A. M. (2011b). Practical aerial grasping of unstructured objects. In 2011 IEEE Conference on Technologies for Practical Robot Applications, 99–104. Woburn, MA.

RamonSoria, P., Arrue, B. C., & Ollero, A. (2017). Detection, location and grasping objects using a stereo sensor on uav in outdoor environments. *Sensors*, *17*(1), 103.

RamonSoria, P., Bevec, R., Arrue, B. C., Ude, A., & Ollero, A. (2016). Extracting objects for aerial manipulation on uavs using low cost stereo sensors. *Sensors*, *16*(5), 700.

Santamaria-Navarro, A., Grosch, P., Lippiello, V., Sola, J., & Andrade-Cetto, J. (2017). Uncalibrated visual servo for unmanned aerial manipulation. *IEEE/ASME Transactions on Mechatronics*, *22*(4), 1610–1621.

Saska, M. (2017). Large sensors with adaptive shape realised by selfstabilised compact groups of micro aerial vehicles. In International Symposium on Robotic Research. Puerto Varas, Chile.

Saska, M., Kratky, V., Spurny, V., & Baca, T. (2017). Documentation of dark areas of large historical buildings by a formation of unmanned aerial vehicles using model predictive control. In 2017 22nd IEEE International Conference on Emerging Technologies and Factory Automation (ETFA), 1–8. Limassol, Cyprus.

Saska, M., Kulich, M., & Preucil, L. (2006). Elliptic net-a path planning algorithm for dynamic environments. *In ICINCO-RA*, 372–377. Setúbal, Portugal.

Spica, R., Franchi, A., Oriolo, G., Bülthoff, H. H., & Giordano, P. R. (2012). Aerial grasping of a moving target with a quadrotor uav. In 2012 IEEE/RSJ International Conference on Intelligent Robots and Systems, 4985–4992. Vilamoura, Portugal.

Tersus-GNSS (2017). PRECIS-BX305 GNSS RTK Board. https://www.tersus-gnss.com. (cited on July 17, 2018).

Thomas, J., Loianno, G., Polin, J., Sreenath, K., & Kumar, V. (2014). Toward autonomous avian-inspired grasping for microaerial vehicles. *Bioinspiration and Biomimetics*, *9*(2), 025010.

Tiderko, A. (2017). multimaster_fkie - Multi-master ROS Package, Robot Operating System (ROS). http://wiki.ros.org/smach. (cited on July 17, 2018).

Weiss, S., Scaramuzza, D., & Siegwart, R. (2011). Monocular-slam-based navigation for autonomous micro helicopters in gps-denied environments. *Journal of Field Robotics*, *28*(6), 854–874.

How to cite this article: Spurný V, Báča T, Saska M, et al. Cooperative autonomous search, grasping, and delivering in a treasure hunt scenario by a team of unmanned aerial vehicles. *J Field Robotics*. 2019;36:125–148.
https://doi.org/10.1002/rob.21816

Chapter 6

Cooperative Transport of Large Objects by a Pair of Unmanned Aerial Systems using Sampling-based Motion Planning

In this chapter, we present the fourth core publication, *Cooperative Transport of Large Objects by a Pair of Unmanned Aerial Systems using Sampling-based Motion Planning* [5c], which was published in the IEEE Conference on Emerging Technologies and Factory Automation in 2019.

[5c] **V. Spurny**, M. Petrlik, V. Vonasek, *et al.*, "Cooperative Transport of Large Objects by a Pair of Unmanned Aerial Systems using Sampling-based Motion Planning," in *IEEE International Conference on Emerging Technologies and Factory Automation (ETFA)*, 2019, pp. 955–962

The **introduction** of the paper explains our motivation to provide the novel methodology required to tackle the problem of cooperative transportation of large objects by a pair of UAVs. This section also lists the contributions of the paper and provides the most related works. To the best of our knowledge, this was the first method enabling motion planning for a UAV pair carrying an object in real time, while also considering the stability of the system in environments with obstacles.

The second section of the paper describes the **system architecture**, which relies on communication to share plans between the UAVs and enables replanning in the event of an appearance of an unknown obstacle. The proposed **motion planning algorithm for cooperative carrying** is explained in the next section. The proposed concept aims to solve the given task by planning a path that prefers to maintain the shape of the pair in a configuration close to the desired one, as long as it is enabled by environmental constraints while allowing for temporary deviation when necessary.

The proposed method in the fourth core publication has been evaluated in numerous simulations and outperformed state-of-the-art sampling-based algorithms in statistical tests in several aspects. The performance and robustness of the entire system were also verified in experiments of autonomous transportation of large objects in an outdoor environment with obstacles. Results from these tests are shown in the **experimental results** section.

This is the author's version of an article that has been published. Changes were made to this version by the publisher prior to publication.
The final version of paper is available at https://doi.org/10.1109/ETFA.2019.8869298

Cooperative Transport of Large Objects by a Pair of Unmanned Aerial Systems using Sampling-based Motion Planning

Vojtech Spurny, Matej Petrlik, Vojtech Vonasek, and Martin Saska
Department of Cybernetics, Faculty of Electrical Engineering
Czech Technical University in Prague
email: vojtech.spurny@fel.cvut.cz

***Abstract*—Cooperative carrying of large, cable-suspended payloads by a pair of cooperating unmanned aerial vehicles (UAVs) is tackled in this paper. The proposed system, involving a sampling-based motion planning algorithm and a model predictive control-based coordination of UAVs, aims to achieve a smooth and reliable flight performance in environments with obstacles. The motion planning is designed to satisfy constraints on relative positions of UAVs, which are defined from the cooperative transport task and by onboard mutual localization, which is used for real-time estimation of states of neighboring robots carrying the object. Besides, a guiding principle with a cost-driven expansion is employed to steer the growth of a Rapidly-exploring Random Tree (RRT) to keep the coupled system of UAVs and the object as close to desired mutual positions as possible. A significant deviation of the controlled system from the desired configuration, by increasing or decreasing the relative distance between UAVs carrying the cable-suspended object, is achieved only if it is required by environment constraints (e.g. in narrow passages), while the allowed limits are always satisfied. Using the guiding principle enables us to find feasible solutions of the problem in a reasonable short time using onboard computer even in environments with a complicated structure of obstacles. The proposed system was evaluated in numerous simulations, compared with state-of-the-art solutions using statistical sets of results, and its performance and reliability were verified in experiments in real-world conditions.**

MULTIMEDIA MATERIAL

A video attachment to this work is available at:
http://mrs.felk.cvut.cz/etfa2019transport

I. INTRODUCTION

Deployment of UAVs for object manipulation, tactile inspection, construction, and other applications that require to interact with the surrounding workspace via external forces between the hovering UAVs and objects in the environment is a very challenging task. This is due to its high complexity, demands on precise control, perception, UAV state estimation, and high risk of collision and mission failure. Although development of such a system is a hot topic in the robotic community (see e.g. results of AEROARMS project[1]), most of the presented solutions have been limited to verification in simplified conditions.

Transport of large object by a single UAV is addressed in [1], where authors are using long reach manipulator consisted from two arms. However, such system requires large and powerful UAV that limits its deployment. Therefore, many authors, including us, are proposing to use multiple UAVs instead of a single one. The most relevant work [2] tackles the transport of objects by rigidly attached quadrotors, but the reliability and performance of this approach is difficult to compare with the proposed solution, since the method is verified only in numerical simulations and the aspects of real-world deployment are not considered. In addition, it requires a known reference trajectory, which limits its usage in autonomous missions. Transport of large objects by multiple UAVs is addressed in [3], [4], where the systems are designed for using in a simple indoor environment without obstacles and a motion capture system is used for team stabilization. A motion capture system was also used in the state-of-the-art solution [5], where stabilization of a bar attached by ropes with different lengths to two UAVs is studied. Their system enables stabilization of the bar in a static position, but it does not allow its transport in environments with obstacles. Transport by UAVs equipped with manipulators in an environment with obstacles is studied in [6]. In this approach, estimate of the UAVs states are obtained by a motion capture system and therefore possible real-world influences are not considered in the motion planning. The proposed approach considers various sources of real-world disturbances and represents an unique solution in terms of applicability in real-world conditions with integrated motion planning technique to allow transportation task in environment with obstacles and narrow passages.

Fig. 1: Examples of manipulation with UAVs.

Motivation for the research aimed to design a complex system being able to solve challenging scenarios of autonomous interaction of cooperating UAVs with objects in real outdoor

[1]https://aeroarms-project.eu/

This is the author's version of an article that has been published. Changes were made to this version by the publisher prior to publication.
The final version of paper is available at https://doi.org/10.1109/ETFA.2019.8869298

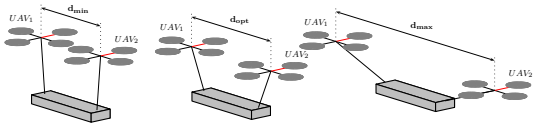


Fig. 2: A pair of UAVs cooperatively carrying a large object. When the UAVs are separated by a minimal distance d_{min} (the left image) the system is vulnerable to external disturbances. If the mutual distance exceeds d_{max} (the right image) the maneuverability of the UAVs is deteriorated due to a lack of thrust, furthermore the gripping mechanism can be disconnected due to an excessive tension in the cables. The optimal mutual distance d_{opt} (the middle image) results in a stable configuration of the payload and acceptable thrust requirements.

scenarios has been identified also in the MBZIRC (Mohamed Bin Zayed International Robotics Challenge) competition[2], which aims to push robotics beyond its current state-of-the art. In MBZIRC 2017, the cooperative objects carrying by a UAV pair was part of the most challenging task - Treasure hunt. Although possibility of transport of large objects using multiple UAVs has been already studied mostly from the control view by the community using cables [7], [8], [9], [10] or rigid connection [3], [11], [12], [4], the difficulty of this task in real-world conditions can be observed in results achieved by the leading robotic laboratories (selected from 143 applications) in the MBZIRC 2017, where none of the competitors was able to solve the given task in a cooperative way.

The system on which we rely in this paper achieved the best score in MBZIRC 2017 and presented the best reliability and robustness, which is crucial for all real-world UAV applications. In the presented work, we go even beyond the MBZIRC competition by extending the applicability of the system into environments with obstacles and by solving a more general problem including motion from an initial state of the object achieved after its cooperative grasping into a desired state (position and orientation). We will focus on the deployment of the system in straitened workspaces with narrow corridors, where the orientation of the object and the relative distance of the UAVs carrying the object have to be changed to enable collision avoidance. This ability together with the possibility of deployment of the system in real-world conditions, both aspects integrated into the motion planning and UAVs coordination, are the main contributions of the presented paper.

The proposed system is modeled and realized as a two-cables-suspended payload, where each UAV is fixed with the object by active grippers mounted at the end of the cables, to allow full maneuverability of both UAVs, which is crucial outdoors in presence of wind and other disturbances. To achieve a stable flight, the distance between the positions of attached grippers on surface of the object must be smaller than the distance between the ends of cables attached to the UAVs in this setup of the system. Based on the system analysis [13] and field experiments, the limits of distances between the UAVs in such a system can be identified together with a desired distance, in which the system is stable and the thrust of both robots is effectively used for reliable flight operations. If the system exceeds the upper limit of the allowed distance, the force induced by the object on the helicopter exceeds its lifting capacity. If the distance between the UAVs is smaller than the lower limit, external disturbances can destabilize the system (see visualization of these limits in Fig. 2). It allows us to formulate the motion planning problem in a space where each state represents the position and orientation of both UAVs, while the limits on minimum and maximum distances between them are satisfied by admissible constraints.

We propose to solve the motion planning problem using a new method based on the Rapidly-exploring Random Tree (RRT) algorithm [14]. The proposed method uses a stochastic optimization with the aim to find a solution in which the states with distance between the UAVs close to the desired one are preferred which increases reliability and efficiency of the cooperative carrying task. Moreover, to be able to find a solution in the straightened workspace with narrow passages, which may block an efficient growth of the tree and significantly prolong the overall computational time [15], [16], we propose to apply ideas of guiding path [17] and cost-driven growth [18], [19] to efficiently solve this planning task. The efficiency of the proposed solution as well as its applicability for the cooperative transport of large objects are shown in numerous statistical analyses, simulations, and experiments in outdoor environment with obstacles in section V.

II. Notation and Problem definition

The configuration of the UAV pair is described by the vector $q = (x_1, y_1, \varphi_1, x_2, y_2, \varphi_2) \in \mathcal{C}$, where (x, y) is the 2D position above the terrain, φ is the heading, and $\mathcal{C}$ is the configuration space. The UAVs are assumed to fly at the constant altitude, which is a necessary assumption due to the applied suspended gripping mechanism. For each UAV, the following simple model can be used: $\dot{x} = v{\cdot}\cos\varphi$, $\dot{y} = v{\cdot}\sin\varphi$, and $\dot{\varphi} = \theta$, where v is the forward velocity and θ is the yaw rate. This motion model guarantees feasible paths that can be accurately tracked by the UAV, since a near-hover flight (without agile maneuvers) is assumed in the cooperative transport scenario.

The geometry of UAVs is modeled by circles for simplification and the obstacles are represented in a known map by a set of 2D polygons. The collision-free configurations, where the UAVs do not touch any obstacle, form the subset $\mathcal{C}_{free} \subseteq \mathcal{C}$. The relative position of the UAVs is constrained to allow a reliable solution of the transportation task. If the distance between the UAVs reaches an upper limit d_{max}, the force induced by the object can exceed their lifting capacity. If the distance between the UAVs is smaller than the lower limit d_{min}, external disturbances can destabilize the system. The set of collision-free configurations in which the distance

[2]http://www.mbzirc.com

This is the author's version of an article that has been published. Changes were made to this version by the publisher prior to publication.
The final version of paper is available at https://doi.org/10.1109/ETFA.2019.8869298

between the two UAVs is larger than d_{min} but less than d_{max}, forms the subset $\mathcal{C}_{valid} \subseteq \mathcal{C}_{free}$. Collision-free paths for both UAVs from the initial configuration $q_{init} \in \mathcal{C}_{valid}$ to the goal configuration $q_{goal} \in \mathcal{C}_{valid}$ are then found in a 6D configuration space under geometric and differential constraints.

III. SYSTEM DESCRIPTION

The system used for stabilization of a UAV pair carrying an object and its navigation into a desired state in environments with obstacles is based on onboard model predictive control technique (MPC) [20] integrated into the control loop for precise following trajectories that can be planned in real-time (see Fig. 3 for a scheme of the entire control and motion planning system). The motion planning algorithm is included in the Motion Planning and Re-planning block, and it is used for simultaneous finding trajectories for both robots in the UAV pair. This block can contain any feasible motion planning technique, which enables to respect the constraints of UAV mutual positions. We propose to use the sampling-based method, which enables to obtain a solution for both UAVs in one planning process. Therefore, the planning task is realized onboard of one of the robots and the obtained trajectories are shared (using Wi-Fi in the presented HW setup) with its neighbor to ensure reliable coordination of the system. The following of the distributed trajectories is achieved using three blocks: MPC Tracker, State Estimation, and Control. The MPC Tracker computes feasible references for the Control unit according to the current information obtained from the State Estimation block (see [21] for details).

In the beginning of the mission, the plan is made with prior knowledge of the environment. For using in real-world conditions, where the known map can differ from reality, the system is able of re-planning in the case of detection of a possible collision. The feasibility check of the current plan is realized in the Collision Warning block, where new obstacles are detected by onboard sensors and added into the map. The relative positions of the UAVs, which are constrained when carrying an object, are obtained either by sharing their global states from an external localization if available (such as GNSS-based state estimation described in [20]) or by a direct onboard mutual localization (for a UV light-based localization purposely designed to solve this task see [22]).

IV. MOTION PLANNING ALGORITHM FOR COOPERATIVE CARRYING

Computing trajectories for multi-UAV systems requires to consider motion models of the UAVs, to avoid their mutual collisions as well as collisions with the obstacles. Moreover, in the case of object carrying, additional constraints have to be considered due to the composed system stability. Sampling-based planners like Probabilistic Roadmaps (PRM) [15] and Rapidly Exploring Random Trees (RRT) [14] are efficient tools to solve this problem as they are able to cope with many-DOF systems and with motion constraints. The idea of the sampling-based planning is to random sample the configuration space $\mathcal{C}$ and classify the samples as free or non-free using collision detection. The free samples are stored in a roadmap and the nearest free samples are connected by edges. Then, the path in the roadmap corresponds to a motion in the workspace.

The basic RRT builds a configuration tree $\mathcal{T}$ rooted at at initial state q_{init} by sub-sequence adding of new reachable feasible configuration. In each iteration of the tree construction, a configuration q_{rand} is randomly sampled from the whole configuration space $\mathcal{C}$ and its nearest neighbor $q_{near} \in \mathcal{T}$ in the tree is found. The configuration q_{near} is then expanded using the motion model to obtain new configurations reachable from q_{near}. The new positions are obtained by applying control inputs to the motion model over time Δt. From these configurations, the nearest one towards q_{rand} is selected and added to the tree. The algorithm terminates if the goal configuration is approached within a given distance or if the maximum number of iterations is reached.

To find paths as fast as possible and also to find path with a good quality in terms of reliability and smoothness of the flight of the composed system, we propose to employ the guiding principle with the cost-driven tree expansion. This extension is crucial, since we aim at design of a system working onboard of UAVs being able to re-plan in case of detected environment alternation, which is necessary in systems operating in real-world conditions. The idea of the guiding-based sampling of the configuration space that we have proposed in [23], [17] is to generate random samples along a predefined guiding path in the workspace. The guiding path can be computed as a simple geometric path, e.g. using Voronoi diagram or Triangular-based methods. The guiding path is represented by n points in a space of lower dimension (in our case 2D) and one of the point is considered as a temporal virtual goal v_{goal}. The random samples are then generated with probability p_v around the virtual goal, otherwise they are generated from the whole configuration space $\mathcal{C}_{valid}$. In our case, the random samples are also generated in 2D space. The distance between the random sample q_{rand} and any configuration of the pair of the UAVs q is computed as the distance between q_{rand} and the middle point between the UAVs in configuration q.

Algorithm 1 shows the pseudocode of the planner using a guiding path, called RRT-path. In each iteration of the RRT-path algorithm, the distance to a temporal virtual goal v_{goal} and the distance to the planner goal q_{goal} is checked. Whenever the distance to a temporal virtual goal v_{goal} is lower then a predefined constant d, the temporal virtual goal is updated to the next point of the guiding path. The tree expansion is interrupted, if the distance to the planner goal q_{goal} is lower than d (a solution is found) or if the maximal number of iterations of the algorithm I_{max} is reached.

In the RRT-path as well as in the basic RRT, the function *expandNode* incrementally expands the tree by applying suitable control inputs that minimize distance to a random sample, which ensures finding a feasible trajectories with respect of dynamic constraints. Unfortunately, this strategy leads to a path that is not smooth enough to provide reliable and effective

This is the author's version of an article that has been published. Changes were made to this version by the publisher prior to publication.
The final version of paper is available at https://doi.org/10.1109/ETFA.2019.8869298

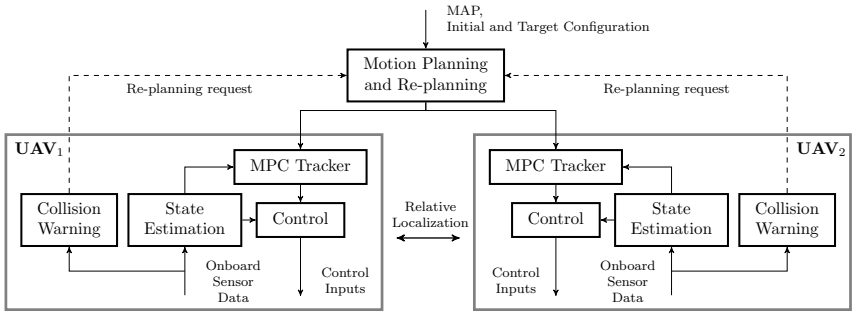


Fig. 3: Diagram of the system pipeline, including the proposed planning method that is in Motion Planning and Re-planning block.

Algorithm 1: RRT-Path planner

input : initial configuration q_{init}
goal configuration q_{goal}
output: $path$
begin
  $\mathcal{T}$.addNode(q_{init});
  v_{goal} = 0;
  p = findGuidingPath();
  for *iteration* = *1:*I_{max} **do**
    if *rand()* $< p_v$ **then**
      q_{rand} = random configuration around $p[v_{goal}]$;
    else
      q_{rand} = random configuration from $\mathcal{C}_{valid}$;
    end
    q_{near} = findNearestNeighbor(q_{rand});
    q_{new} = expandNode(q_{near}, q_{rand});
    if $q_{new} \neq$ *NULL* **then**
      $\mathcal{T}$.addNode(q_{new});
    end
    if *distance to* $v_{goal} < d$ **then**
      update v_{goal};
    end
    if *distance to* $q_{goal} < d$ **then**
      break;
    end
  end
  $path$ = reconstructPath($\mathcal{T}$, q_{goal});
  return $path$;
end

motion of the system in real-world conditions since it contains frequent changes of the control inputs, which may result system oscillations. We propose to employ a novel cost-driven tree expansion design to solve the cooperative carrying task. In the expansion of the tree, control inputs with the minimal value of a cost function, which penalizes their changes, are selected.

The path that is smooth and where the UAVs are maintaining the ideal bounded mutual distance between themselves for the most of the time is preferred in the scenario of cooperative large object transportation. To achieve this requirement, the cost of the configuration q, reached by expanding the tree from the configuration q_{near} toward the configuration q_{rand}, is obtained as

$$\begin{aligned} c(q, q_{near}, q_{rand}) = \alpha \cdot (d(q) - l)^2 \\ + \beta \cdot dist(q, q_{rand}) \\ + \gamma \cdot h(q_{near}, q). \end{aligned} \quad (1)$$

The first part of the cost function, where $d(q)$ is distance between UAVs at configuration q and l is the ideal distance, is designed to penalize configurations in which the mutual distance differs from the ideal one. The second part, where $dist(\cdot, \cdot)$ represents the Euclidean distance between two configurations, is included to prefer configurations which are closer to the random sample. The aim of the last part is to prefer solutions, in which the movement direction remains unchanged in consequent steps. The value of function $h(q_{near}, q)$ is minimal (0) in case the tree is expanded into the same direction as it was for q_{near} and maximal (1) if the tree is expanded into the opposite direction. Only the control inputs that lead to a valid configuration $q \in \mathcal{C}_{valid}$ are used to grow the tree. Parameters α, β, and γ influence type of the preferred path. For example by increasing only the parameter α (with the remaining parameters unchanged), configurations of UAVs that have mutual distance closer to the ideal distance will be preferred at the cost of relaxing the requirements on path smoothness and distance from q_{rand}.

V. EXPERIMENTAL RESULTS

Two sets of experiments have been realized to evaluate the RRT-path cost-driven (RRTc-path) planner and to verify functionality of the proposed system.

A. RRTc-path versus existing methods

The proposed RRTc-path algorithm has been compared with the basic RRT, RRT-path, and Transition-based RRT (T-RRT) [19] methods. The generation of control inputs for an expansion of a node in the tree is achieved by combinations of the forward and yaw rate controls of both UAVs. For these experiments, only forward or no movement of the UAVs is allowed ($v \in \{0, 1\}$ $[m/s]$, $\theta \in \langle -75, 75 \rangle$ $[deg/s]$, and $\Delta t = 1$ $[s]$) and the branching factor of a node in the tree

is 195 (the yaw with 13 values). The maximal and minimal allowed mutual distances have been identified as $d_{max} = 4$ $[m]$ and $d_{min} = 2$ $[m]$, respectively. The required ideal mutual distance for transport of a large object is 3 m, i.e. $d_{opt} = 3$ $[m]$. The parameters for particular algorithms are set as

- RRTc-path: the parameters of the cost function (eq. (1)) - $\alpha = 10$, $\beta = 1$, and $\gamma = 3$.
- T-RRT: $nFail_{max} = 10$, $\alpha = 2$, $cost(q) = (d(q) - l)^2$.

The same cost function cannot be used for both T-RRT and RRTc-path techniques. In the T-RRT method, a node that is the closest one to the randomly generated sample is selected and added to the tree if a transition test is passed. However, the distance to the randomly generated sample is already included in the cost function that we propose. Therefore, it was necessary to use a modified cost function $cost(q) = (d(q)-l)^2$ for the T-RRT experiments to achieve comparable results. Since the cost functions of the T-RRT and RRTc-path planners are different and the basic RRT and RRT-path techniques do not allow using a cost function, we present a comparison of the individual quality criteria important for the application instead of the total cost of found solution.

The performance of all planners has been tested in an environment without obstacles and in an environment with several walls including a narrow passage (Fig. 4). Obstacles are dilated by the dimension of the UAV to prevent collisions. The quality of achieved solutions is compared by

- the length of the path (average from paths of both UAVs),
- the smoothness of the path (sum of the $h(\cdot, \cdot)$ function (see eq. (1)) along the entire path normalized by its length),
- average deviation from the ideal mutual distance between UAVs,
- maximal deviation from the ideal mutual distance between UAVs,
- and the number of iterations of the algorithm.

The obtained results are summarized in Table I. All performance results in Table I, except for the maximal deviation from the ideal mutual distance between UAVs, correspond to average values 20 runs with the same setting.

The growth of the tree in the RRT-path and the proposed RRTc-path techniques is stimulated by the guiding principle. Due to this technique, shorter solutions compared to the RRT or T-RRT algorithm tend to be found. Additionally, these solutions are found using significantly lower number of iterations. All planners except for the RRTc-path do not consider requirements on smoothness of the solution, which results in higher values of the smoothness indicator (the first column in the table) in comparison with the RRTc-path. If only these three planners are compared, the RRT-path planner provides solutions that are more smooth than the RRT and T-RRT because the tree expansion in this case is not entirely randomized into all direction but around the guiding path.

Experiments clearly confirm that the T-RRT and RRTc-path methods enable to find solutions with significantly smaller deviation from the ideal mutual distance between the UAVs compared to the RRT and RRT-path, due to the applied cost function. The T-RRT algorithm uses the transition test as an evaluation criteria for the quality of a new configuration [19]. A new configuration q_{new}, acquired by applying selected control inputs from the state q_{near}, is automatically accepted by this test whenever the cost of q_{new} is less or equal than the cost of q_{near}. Therefore, if the distance between the UAV pair is deflected from the required distance, the tree is rarely expanded into a configuration with smaller deviation from the ideal mutual distance and the system remains deflected. This behavior of the T-RRT method is visible in Table II, where results of sets of 20 experiments in the environment without obstacles are presented. In these experiments, UAVs are purposely initiated in configurations, in which the mutual distance between the UAVs differs by 0.5 m from the required distance $d_{opt} = 3$ $[m]$. The T-RRT algorithm provides solutions in which the UAVs transport an object in a formation in almost the same deviated mutual distance along the entire trajectories, since there is no mechanism to prioritize better configurations. The proposed RRTc-path method provides solutions in which the UAV pair maintains the ideal distance whenever it is enabled by environmental constraints (e.g. if the formation is not forced to shrink its shape by obstacles in narrow passages).

	smooth. [m]	**max. dev. [-]**	**ave. dev. [m]**	**length [m]**	**iter. [-]**
empty space					
· RRT	0.184	0.845	0.150	137.800	1362
· T-RRT	0.184	0.156	0.015	135.800	1016
· RRT-path	0.061	0.925	0.429	**112.800**	**116**
· **RRTc-path**	**0.001**	**0.096**	**0.001**	112.900	129
narrow corridor					
· RRT	0.190	1.000	0.421	239.025	2576
· T-RRT	0.196	0.402	0.056	240.900	1672
· RRT-path	0.067	1.000	0.610	**180.875**	**193**
· **RRTc-path**	**0.015**	**0.218**	**0.025**	204.625	697

TABLE I: Comparison of results obtained by solving the cooperative motion planning problem in the environments in Fig. 4. The most important properties for stable and robust cooperative transportation, smoothness of the obtained paths and mutual UAVs position close to the optimal formation, are evaluated in the first three columns. Although the minimal length of the paths (the fourth column) is also a desired property, it is not the most important for the given application. The number of iteration (the fifth column) influences computational requirements. Nevertheless, all evaluated algorithms provided the result in a reasonable time using onboard computer in these maps.

B. Real-world Experiments

The overall ability of the system to achieve cooperative transportation of a large object by a UAV pair has been verified in numerous experiments in real outdoor conditions. Hexacopters with DJI F550 frames and PixHawk low-level stabilization have been used carrying NUC onboard PC and suspended gripping mechanism (see [24] for details on the

This is the author's version of an article that has been published. Changes were made to this version by the publisher prior to publication.
The final version of paper is available at https://doi.org/10.1109/ETFA.2019.8869298

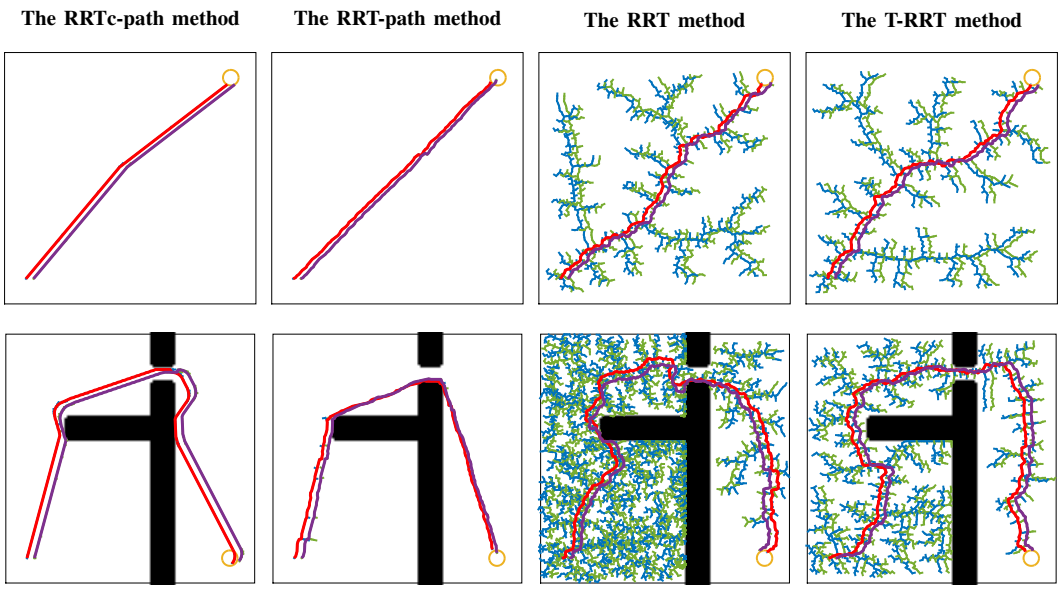


Fig. 4: Comparison of results obtained by the proposed RRTc-path method with three different existing techniques. An example of solution paths and trees after the search processes in an environment without obstacles is shown in the first row, and in environment with a narrow passage in the second row. The solution paths connect initial configurations (left down corner of the maps) and goal configurations found near to the goal position located at opposite corner of the environment (visualized by yellow circles). The blue segments display the paths in the tree for the first UAV and the green segments for the second UAV. The solution paths are denoted by the red and purple segments.

	deviated initial configuration		perfect initial configuration	
	max. dev. [m]	**ave. dev. [m]**	**max. dev. [m]**	**ave. dev. [m]**
T-RRT	0.514	0.463	0.156	0.015
RRTc-path	0.500	0.024	0.096	0.001

TABLE II: Experiments with a formation of UAVs purposely initiated with a mutual distance, which differs by 0.5 m from the required distance $d_{opt} = 3$ $[m]$. Results were obtained by different algorithms in the empty environment shown in the first row of Fig. 4.

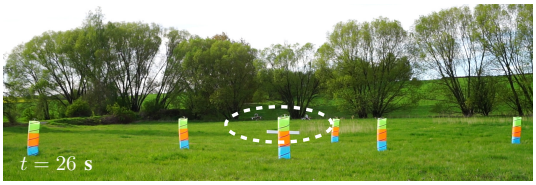
device designed for the MBZIRC 2017 competition) to realize the cooperative carrying task.

In the experiments, a UAV pair is employed to transport an object through a field with cylindrical obstacles shown in Fig. 5 and through a narrow passage in Fig. 6 without collision with the obstacles. Graphs showing a record of collision-free paths flown by the UAV pair are in Fig. 7. As shown in Fig. 8, limits on mutual distances between the UAVs, $d_{min} = 2$ $[m]$ and $d_{max} = 4$ $[m]$, have been satisfied in both experiments. In the real-world experimental deployment, it was observed that the UAV pair motion was influenced by sensory noise and external disturbances even if using well tuned controllers, which achieve position accuracy less than 5 centimeters in ideal conditions (see the results in [20]). The motion oscillation caused by not perfectly precised sensory information, real actuators and presence of wind that effects on the large surface of the carried object can be seen in Fig. 8. The experimental deployment also confirmed the need of the presented motion planning approach, which assumes that the UAVs do not follow the given plan perfectly and that any strong disturbance can destabilize the formation coupled by the object if it operates close to operational limits. Such an observation during the initial experiments was also a motivation to design this motion planner and it would not be possible if relying on simplified laboratory tests using motion capture systems only.

VI. CONCLUSIONS

In this paper, a system designed for solving a cooperative transportation task by a UAV pair in environments with obstacles was presented. The proposed concept aims to solve the given task by planning a path that prefers to maintain the shape of the pair in a configuration that is close to the desired one as long as it is enabled by environmental constraints, while allowing for temporary deviation when necessary. This enables to prevent the pair carrying the object from undesired oscillations

This is the author's version of an article that has been published. Changes were made to this version by the publisher prior to publication.
The final version of paper is available at https://doi.org/10.1109/ETFA.2019.8869298



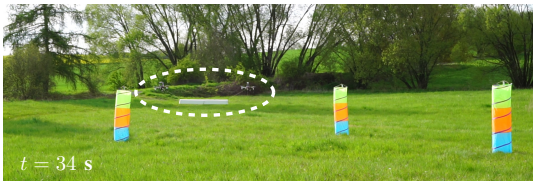


Fig. 5: Snapshots from the outdoor experiment, in which a large object is transported through the field with cylindrical obstacles by a UAV pair. The pair is highlighted by the white dashed ellipse.

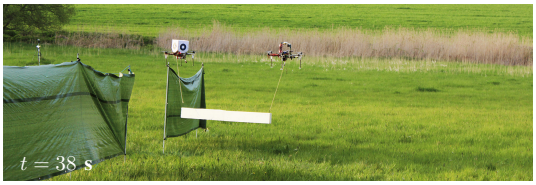


Fig. 6: Snapshots from the outdoor experiment, in which a large object is transported through a narrow passage.

and increase robustness of the system in real-world conditions, where sensory data are effected by noise and UAVs may be subject to external disturbances. The core of the presented solution is a sampling-based motion planning algorithm, which was purposely designed to satisfy these needs. The novel method takes advantage of a guiding-based sampling of the configuration space, while in the same time includes cost-driven expansion to find an appropriate solution of the given task in terms of smoothness and quality of achieved paths with respect of stableness and reliability. Due to the cost involved in the planning process, most of the mission the UAVs are forced to operate in relative positions that were identified as the most proper, but the system enables to temporarily deviate from this desired shape of the formation in case it is required by surrounding environment. The guiding-based process is important to fasten the motion planning task and to offer an ability of fast onboard re-planning in case of detected environment alternation, which is also important for reliable real-world deployment of the system. The proposed method has been evaluated in numerous simulations and outperformed state-of-the-art sampling-based algorithms in statistical tests in several aspects. Beyond the object carrying application, the comparison with currently used approaches presented a high application potential of the proposed RRTc-path method in other multi-UAV tasks and motion planning problems, where computational time and quality of the obtained solution with respect of a given criteria is important to achieve. Finally, the performance and robustness of the entire system were verified in experiments of autonomous transportation of large objects in an outdoor environment with obstacles in different weather conditions. According to our knowledge, it is the first solution of the autonomous cooperative transportation task in outdoor environment with obstacles that achieves sufficient reliability in real-world conditions such as in presence of wind.

ACKNOWLEDGMENT

The work has been supported by CTU grant no. SGS17/187/OHK3/3T/13, by the Grant Agency of the Czech Republic under grant no. 17-16900Y, and by Research Center for Informatics project CZ.02.1.01/0.0/0.0/16_019/0000765.

REFERENCES

[1] A. Caballero, A. Suarez, F. Real, V. M. Vega, M. Bejar, A. Rodriguez-Castaño, and A. Ollero, "First experimental results on motion planning for transportation in aerial long-reach manipulators with two arms," in *IEEE IROS*, 2018, pp. 8471–8477.

[2] Z. Wang, S. Singh, M. Pavone, and M. Schwager, "Cooperative object transport in 3d with multiple quadrotors using no peer communication," in *IEEE ICRA*, 2018, pp. 1064–1071.

[3] D. Mellinger, M. Shomin, N. Michael, and V. Kumar, *Cooperative Grasping and Transport Using Multiple Quadrotors*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2013, pp. 545–558.

[4] R. Ritz and R. D'Andrea, "Carrying a flexible payload with multiple flying vehicles," in *IEEE IROS*, 2013, pp. 3465–3471.

[5] P. O. Pereira, P. Roque, and D. V. Dimarogonas, "Asymmetric collaborative bar stabilization tethered to two heterogeneous aerial vehicles," in *IEEE ICRA*, 2018, pp. 5247–5253.

[6] H. Lee, H. Kim, W. Kim, and H. J. Kim, "An integrated framework for cooperative aerial manipulators in unknown environments," *IEEE RAL*, vol. 3, no. 3, pp. 2307–2314, 2018.

[7] J. Fink, N. Michael, S. Kim, and V. Kumar, "Planning and control for cooperative manipulation and transportation with aerial robots," *The International Journal of Robotics Research*, vol. 30, no. 3, pp. 324–334, 2011.

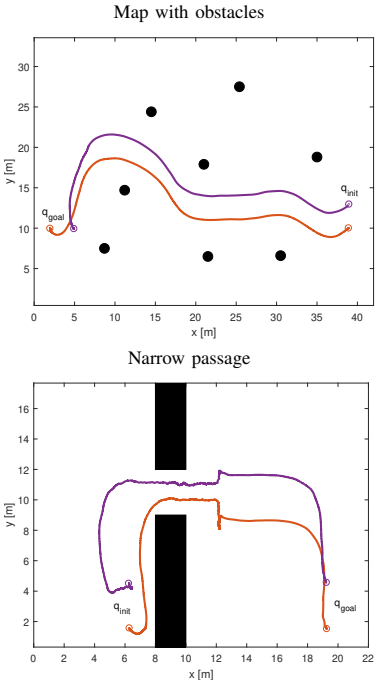


Fig. 7: A record of paths passed by the pair of UAVs during the hardware experiments. The top image shows paths from the experiment in environment with cylindrical obstacles (the black dots in the image) presented in Fig. 5. Data from the experiment in Fig. 6, in which an object was transported by the UAV pair through a narrow passage, are presented in the bottom image.

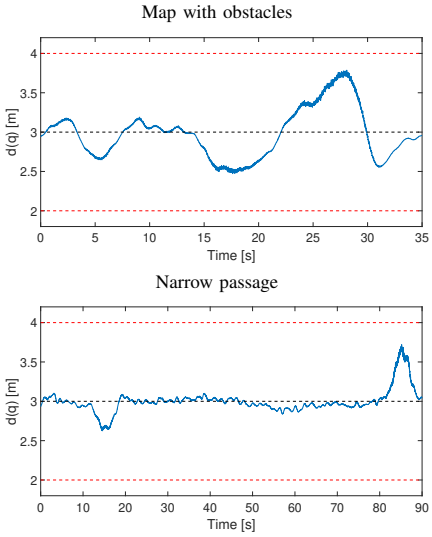


Fig. 8: Mutual distance between the UAVs while transporting an object in the experiments. The top image shows data from the experiment presented in Fig. 5, while the bottom image data from the experiment in Fig. 6. The optimal mutual distance is depicted by the black line. The red dashed lines represent the maximal allowed deviation from the optimal mutual distance.

[8] M. Gassner, T. Cieslewski, and D. Scaramuzza, "Dynamic collaboration without communication: Vision-based cable-suspended load transport with two quadrotors," in *IEEE ICRA*, 2017, pp. 5196–5202.
[9] Z. Li, J. F. Horn, and J. W. Langelaan, "Coordinated transport of a slung load by a team of autonomous rotorcraft," in *AIAA Guidance, Navigation, and Control Conference*, 2014, p. 968.
[10] M. Bernard, K. Kondak, I. Maza, and A. Ollero, "Autonomous transportation and deployment with aerial robots for search and rescue missions," *Journal of Field Robotics*, vol. 28, no. 6, pp. 914–931, 2011.
[11] V. Parra-Vega, A. Sanchez, C. Izaguirre, O. Garcia, and F. Ruiz-Sanchez, "Toward aerial grasping and manipulation with multiple uavs," *Journal of Intelligent & Robotic Systems*, vol. 70, no. 1, pp. 575–593, Apr 2013.
[12] G. Loianno and V. Kumar, "Cooperative transportation using small quadrotors using monocular vision and inertial sensing," *IEEE RAL*, vol. 3, no. 2, pp. 680–687, 2018.
[13] M. Tognon, C. Gabellieri, L. Pallottino, and A. Franchi, "Aerial co-manipulation with cables: The role of internal force for equilibria, stability, and passivity," *IEEE RAL*, vol. 3, no. 3, pp. 2577–2583, 2018.
[14] S. M. LaValle, "Rapidly-exploring random trees: A new tool for path planning," 1998, technical report 98-11.
[15] L. E. Kavraki, P. Svestka, J.-C. Latombe, and M. H. Overmars, "Probabilistic roadmaps for path planning in high-dimensional configuration spaces," *IEEE Transactions on Robotics and Automation*, vol. 12, pp. 566–580, 1996.
[16] D. Hsu, J.-C. Latombe, and R. Motwani, "Path planning in expansive configuration spaces," in *International Journal of Computational Geometry and Applications*, vol. 3, 1997, pp. 2719–2726.
[17] V. Vonásek, J. Faigl, T. Krajník, and L. Přeučil, "RRT-path — a guided rapidly exploring random tree," in *RoMoCo*. Springer, 2009, pp. 307–316.
[18] S. Karaman and E. Frazzoli, "Sampling-based algorithms for optimal motion planning," *The International Journal of Robotics Research*, vol. 30, no. 7, pp. 846–894, 2011.
[19] L. Jaillet, J. Cortés, and T. Siméon, "Transition-based rrt for path planning in continuous cost spaces," in *IEEE IROS*, 2008, pp. 2145–2150.
[20] V. Spurny, T. Baca, M. Saska, R. Penicka, T. Krajnik, J. Thomas, D. Thakur, G. Loianno, and V. Kumar, "Cooperative autonomous search, grasping, and delivering in a treasure hunt scenario by a team of unmanned aerial vehicles," *Journal of Field Robotics*, vol. 36, no. 1, pp. 125–148, 2019.
[21] T. Baca, D. Hert, G. Loianno, M. Saska, and V. Kumar, "Model predictive trajectory tracking and collision avoidance for reliable outdoor deployment of unmanned aerial vehicles," in *IEEE IROS*, 2018.
[22] V. Walter, M. Saska, N. Staub, and A. Franchi, "Mutual localization of uavs based on blinking ultraviolet markers and 3d time-position hough transform," in *IEEE CASE*, 2018.
[23] V. Vonasek, "A guided approach to sampling-based motion planning," Ph.D. dissertation, CTU, Czech Technical University in Prague, 2016. [Online]. Available: https://dspace.cvut.cz/handle/10467/62503
[24] G. Loianno, V. Spurny, J. Thomas, T. Baca, D. Thakur, D. Hert, R. Penicka, T. Krajnik, A. Zhou, A. Cho, M. Saska, and V. Kumar, "Localization, grasping, and transportation of magnetic objects by a team of mavs in challenging desert-like environments," *IEEE RAL*, vol. 3, no. 3, pp. 1576–1583, 2018.

Chapter 7

Documentation of Dark Areas of Large Historical Buildings by a Formation of Unmanned Aerial Vehicles Using Model Predictive Control

This section presents the fifth core publication called *Documentation of Dark Areas of Large Historical Buildings by a Formation of Unmanned Aerial Vehicles using Model Predictive Control* [6c] published in the IEEE Conference on Emerging Technologies and Factory Automation in 2017. This publication was awarded the best conference paper.

[6c] M. Saska, V. Kratky, **V. Spurny**, *et al.*, "Documentation of Dark Areas of Large Historical Buildings by a Formation of Unmanned Aerial Vehicles using Model Predictive Control," in *IEEE International Conference on Emerging Technologies and Factory Automation (ETFA)*, 2017, pp. 1–8

The **introduction** of the paper highlights the contribution of this work in the robotic context and in regards to deployment of UAVs for the documentation of historical buildings. During the time of publication, we found only the work [84] that uses UAVs in the context of documenting the interiors of historical buildings. In order to obtain a 3D model of the building, their method is based on an online visual simultaneous localization and mapping and an offline visual structure-from-motion method. Our method goes much further and aims to fully exploit the abilities of UAVs, which is primarily flying in areas that are difficult to access for humans and other types of robots. We propose using a formation of cooperating UAVs for visual the documentation and inspection of the interiors of historical buildings. Still, the contribution goes beyond the context of the application. It provides a theoretical work on multi-objective planning for teams of UAVs that are able to incorporate motion constraints with requirements of general mission objectives while considering safety in real-world conditions, as is the main focus of this thesis.

The contributions of this core paper are collected in the section **motion planning and formation stabilization system**. The experimental verification of the proposed methodology was done in the realistic robotic simulator. The results from this simulated experiment and demonstration of the system in the real-world are shown in the section **experimental verification and system deployment**.

This is the author's version of an article that has been published. Changes were made to this version by the publisher prior to publication.
The final version of paper is available at https://doi.org/10.1109/ETFA.2017.8247654

Documentation of Dark Areas of Large Historical Buildings by a Formation of Unmanned Aerial Vehicles using Model Predictive Control

Martin Saska and Vít Krátký and Vojtěch Spurný and Tomáš Báča
Department of Cybernetics, Faculty of Electrical Engineering
Czech Technical University in Prague
email: martin.saska@fel.cvut.cz, tel: +420224357634

***Abstract*—A system designed for a unique multi-robot application of closely flying formations of Unmanned Aerial Vehicles (UAVs) in indoor areas is described in this paper. The proposed solution is aimed as a tool for historians and restorers working in large historical buildings such as churches to provide an access to areas that are difficult to reach by humans. In these objects, it is impossible to keep a large scaffolding for a long time due to regular services, which is necessary for studying a long-term influence of restorations works, and some parts of the churches were even not reached by people for decades and need to be inspected. To provide the same documentation and inspection techniques that are used by the experts in lower easily accessible parts of the buildings, we employ a formation of autonomous UAVs, where one of the robots is equipped by a visual sensor and the others by source of light, which provides the required flexibility for control of lightening.**

The described system in its full complexity has been implemented with achieved robustness and reliability required by deployment in real missions. The technology demonstration has been provided with real UAVs in historical objects to help restorers and conservationists with achieved valuable results used in plans of restoration works. In these missions, UAVs were autonomously hovering at designated locations to be able to demonstrate usefulness of such robotic lightening approach.

I. Introduction

A system for autonomous documentation of dark areas in large historical buildings by a formation of Unmanned Aerial Vehicles (UAVs) is presented in this paper. In the proposed approach, a self-stabilized formation of multi-rotor helicopters is employed for filming and visual inspection in dark conditions, where one of the UAVs carries a camera and neighboring UAVs a source of light. This setup aims to fully autonomously realize two techniques often used by historians and restorers for manual inspection of interiors of historical monuments nowadays. The first one, so-called *Three point lighting* approach [1], [2], is a filming technique in which 1-3 sources of light are used in different locations relatively to the camera optical axis. The method enables to create an illusion of a three-dimensional subject in a two-dimensional image and to illuminate the subject being shot (such as sculptures in historical buildings) while controlling the shading and shadows produced by lighting. This is essential for the presentation of historical monuments in interiors to the broad public, as it removes the boring flatness from images and videos, and it adds a value to the analysis of gathered results by historians.

The second technique frequently used by restorers employs a strong side-light for illumination of flat objects, such as walls with parget and mosaics. In this method, the strong light needs to be placed as close as possible to the scanned plain, which makes visible shadows in the image in a case of a roughness of the surface. Restorers and conservationists can detect from such illuminated pictures if a tile in the mosaic is not fixed properly or if a painting is affected by a humidity indicated by buckling of the wall surface.

We implemented these two methods using autonomous unmanned helicopters, which enables their usage in places in the interior of large historical objects (e.g. churches) that are not accessible without installation of a costly staging. In cooperation with restorers and filmmakers, we defined the filming *Three point lighting* technique and the *Strong side-lightening* method as a multi-objective optimization problem in the Model Predictive Control (MPC) framework. The MPC approach is used for control of the formation members taking into account task objectives as well as constraints of the formation flying (obstacle avoidance, mutual collision avoidance), low level UAV stabilization [3] (motion constraints), filming (limited camera field of view, keeping all UAVs out of the taken images), and illumination (providing intensity of the lightening in a required range, keeping the recommended angle between the light and camera axes).

Fig. 1. Demonstration of the presented formation flying approach to realize a flying "film crew" in a church in Sternberk, Moravia.

It is not our intention to design a system that would decide where and what to film or scan. This is a job of the filmmakers and experts from the field of restoring and historical sciences.

CHAPTER 7. DOCUMENTATION OF DARK AREAS OF LARGE HISTORICAL BUILDINGS BY A FORMATION OF UNMANNED AERIAL VEHICLES USING MODEL PREDICTIVE CONTROL

This is the author's version of an article that has been published. Changes were made to this version by the publisher prior to publication.
The final version of paper is available at https://doi.org/10.1109/ETFA.2017.8247654

To be able to reach the direct usability of the system by the experts, we employ the artificial intelligent and the autonomy of UAVs only for solving technical tasks, such as the lightening, and the artistic intent is left at hands of the professionals. Also the decision, which part of the church needs to be inspected and which technique has to be used is still taken by historians and the proposed system is designed only as a smart tool to facilitate their job. Therefore, we assume an input from these experts for the system in a form of a given path that has to be followed by the UAV with a camera and a given set of objects that have to be filmed. From the safety reason, we expect a given map and known positions of the objects of interest in the building. Although, online mapping techniques are provided by robotic community nowadays, they lack 100% reliability, and also filmmakers as well as historians require visualisation of the planned path prior the mission to make sure that their intention will be fulfilled for which the map given prior the mission is essential. In addition, the revision of the overall plan increases probability of mission failure, which is crucial for deployment of autonomous systems flying close to objects with high historic and financial costs. Fortunately, in the presented application scenario, the knowledge of a precise map prior the mission is a realistic assumption since the precise map is always used by the historians and it is a main part of the restoration and historical survey for which these techniques are targeted.

In practice, the experts denote which part of the path has to be dedicated to which object of interest (OoI) and the camera orientation is simply computed for each point of the path using the known position of the OoI. The obtained profile of the camera heading does not need to be smooth and feasible since it is considered only as one of the desired objectives of the MPC optimization. Usually, the task requirements do not respect the UAV movement constraints, mainly if the input for the system is provided by the historians. The film lighting technicians have usually better understanding of motion constraints of a camera doing a dynamic shot, but still their demands are often in contrast with limitations of UAVs and self-localized formations. Therefore, the proposed system is designed to find a solution close to the usually infeasible (and even non-continuous) desired trajectory for the leader and the camera orientation, and to find a compromise between the requirements given by the experts and the artificial system of closely cooperating UAVs. Besides, we expect that the positions of the objects of interest, obstacles and neighbouring robots may not by known absolutely precisely, and the planning and control of the multi-robot team is then driven by actual results of UAV perception (obstacle detection and object recognition). In the current implementation, the solution is prepared for using the system of neighboring objects/UAVs localization that we have described in [4], [5], but any state-of-the-art implementation of proper obstacle detection and object recognition techniques can be used.

A. Literature review

Autonomous systems used for documentation of heritage sites are mostly employed for 3D modelling of exteriors and interiors of objects by laser scanners or by the photogrammetry, which is being popular recently as it allows to use cheap and lightweight monocular cameras for 3D object reconstruction. Numerous techniques exist to facilitate and speed up the scanning process by increasing its automatization. Static terrestrial laser scanners were used in [6], [7] for 3D modelling of historical sites. Using these techniques, the scanners have to be placed into predefined geo-referenced locations and the obtained 3D point cloud is fused using the known positions of different measurements. In [8], a long-range 3D laser scanner was used to scan large historical sides and to provide an inter-relationship between outdoor and indoor profiles using a technique for storing and processing big amount of data. A faster scanning process is allowed by a hand-held mobile mapping system called Zebedee [9] that was designed for semi-autonomous gathering of 3D point cloud models in cultural heritage applications. Another level of autonomy has been added using a ground mobile robot carrying the laser scanner that enables 3D modelling of large-scale environments with minimal human intervention [10].

Applying UAVs provides a possibility to acquire data from measurement locations inaccessible by ground robots or hand-hold systems in addition to the autonomy provided by UGVs. Nowadays, UAVs are deployed mainly for 3D modelling of outdoor archaeological locations by the photogrammetry of geo-referenced images and historical buildings by lightweight lasers [11], [12], [13], [14]. All these techniques, which are offered by numerous private companies, use GPS data to reference the images and to initialize the 3D cloud construction process by this information. Only a limited number of approaches allows 3D modelling by UAVs in GPS-denied environment inside historical buildings, although indoor localization and 3D mapping techniques are well investigated by robotic community [15], [16]. We have found only the work in [17] that uses UAVs in the context of documentation of interiors of historical buildings. The method is based on a state-of-the-art online visual simultaneous localization and mapping and an offline visual structure-from-motion method in order to obtain the 3D model of the building.

Our method goes much further than [17] and it aims to fully exploit the abilities of UAVs, which is mainly flying in places hardly accessible by people and other types of robots. It was identified during the first deployment of the system in large churches that in these hardly accessible locations light conditions are often not sufficient and external light sources are required. The proposed system enables to set direction of the light sources dependently on the position of the camera, which is a very useful tool being used by restorations and filmmakers. This is realised by a leader-follower formation flying technique providing the *Three point lighting* and *Strong side-lightening* approaches in an autonomous way.

The proposed formation stabilization and navigation algo-

CHAPTER 7. DOCUMENTATION OF DARK AREAS OF LARGE HISTORICAL BUILDINGS BY A FORMATION OF UNMANNED AERIAL VEHICLES USING MODEL PREDICTIVE CONTROL

This is the author's version of an article that has been published. Changes were made to this version by the publisher prior to publication.
The final version of paper is available at https://doi.org/10.1109/ETFA.2017.8247654

rithm, which arises from our theoretical work on formation stabilization of ground robots [18], [19], [20] and UAVs [21], [22], [23], [24], consists of two main components. In the first one, an optimization-based model predictive control mechanism is employed for the formation leader, which is the UAV carrying the camera. A sequence of control inputs in the receding horizon fashion is provided for the leader in two optimization steps. Firstly, a multi-objective optimization is used to control the 3D position of the robot along the predefined path. Deviation from the desired path that is given by the restorers, too aggressive control inputs (changing velocity and angular acceleration), and too small distance from the obstacles and the objects of interest are penalised in this optimization. The second optimization step of the planning method proposed for the leader is aimed to provide a smooth and feasible profile of the camera orientation. Motion planning and stabilization of followers are also realized under the model predictive control scheme in two optimization processes to control their position within the formation and to control orientation of their light sources. In each control step, a new control sequence is obtained as a result of the multi-objective optimization procedures, which is applied in the receding horizon fashion. It means that in each step, only a small portion of the plan is sent to the actuators and for the consequent re-planning a new piece of the same size is added at the end of the plan.

II. Preliminaries

A. Task definition and assumptions

The multi-robot task solved in this paper is to realize the *Three point lighting* and *Strong side-lightening* approaches in indoor GPS-denied environment by a team of cooperating UAVs (multi-rotor helicopters). In this scenario, one of the UAV (the leader) is equipped with a calibrated camera and n_F UAVs (the followers) with light sources. Parameters of the camera and light sources (orientation, field of view, light intensity, dispersion, etc.) are known prior the mission.

In the description of the proposed method, we suppose a known map of the environment (a contour of walls) with a set of convex obstacles O. Although no dynamic obstacles usually occur in the real deployment of our approach for documentation of historical buildings, the method in general is able to deal with a dynamic environment and with an unknown or partial known map as we have shown in a theoretical work in [25]. We also suppose that each UAV is equipped with a system providing its localization in the given map and relatively to its neighbours. For the real deployment in GPS-deneid environment, we propose to rely on a fusion (the mechanism of the fusion is described in [26]) of an odometry obtained from an optical flow in images captured by a down-looking camera and outputs of visual relative localization of neighbours (for description of the employed localization method see [4], [5]). In the case of the *Strong side-lightening* technique, we assume to have a precise information on the distance to the scanned wall. A communication link between the robots is assumed during the mission for distribution of the actual leader's plan to followers, which is used for planning their optimal positions and orientations of the light sources in the future.

As mentioned, our intention is not to contribute to the art and to jobs of the restorers and we suppose that a desired path for the leader is given by these experts. In addition, we suppose that the path is labelled by an information which OoI has to be observed in which part of the path. Using the information on known positions of OoI, the desired orientation of the camera may be simply derived. The desired path and camera orientation given by the experts, later denoted by the $(\cdot)^{ex}$ symbol, does not need to be feasible with respect of kinematics of UAVs.

Let us note that the proposed method is capable of working with unknown a priori knowledge of positions of obstacles and objects of interest (they can be detected by state-of-the-art methods such as the one we proposed in [5]) and work in a full perception driven mode. Also the given desired path is not necessary for the proposed formation flying method and a trajectory planning method can be integrated as in our previous work in [21]. Nevertheless, such a higher level of autonomy is not required in this application and it adds an additional source of uncertainties, which would decrease robustness and so the applicability of the method.

B. Model predictive control on a receding horizon

MPC-based approach is used in the proposed system to independently control the position of the leader as well as the followers in the formation, but also to control the orientation of the carried camera and the light sources to enable stability analyses of the complete system. In the MPC scheme, the control task is transformed into an optimization with constraints imposed by a model of the controlled system. We will use the kinematic UAV model from [21] for both, the leader and the followers. The optimization solves a finite horizon optimization control problem starting from current state over a control horizon. The result of the MPC method is then an optimal trajectory defined by N transition points and constant control inputs between them. The optimization is initialized in each control step based on the solution obtained in the previous MPC step. In each MPC step, once a solution of the optimization problem is obtained, only the first n vectors of the computed control inputs are applied to control the system. The optimization process is then repeated on a new interval as the finite horizon moves by the time horizon with length $n\Delta t$, where Δt is the time difference between two subsequent transition points.

III. Motion planning and formation stabilization system

In this section, a formation flying method based on the MPC framework employed for control and navigation of an autonomous team of a cameraman and illuminators will be described. In the method description, the more general setup of a camera and lights mounted on a gimbal is used. This enables to control the Yaw angle independently to the task, which

This is the author's version of an article that has been published. Changes were made to this version by the publisher prior to publication.
The final version of paper is available at https://doi.org/10.1109/ETFA.2017.8247654

enables a very important safety mechanism. If all UAVs are oriented in the direction of their movement along the desired trajectories (independently to positions of the OoI and so the camera/gimbal orientation), it is much simpler to manually overtake their control in a case of a system malfunction. This is important mainly during the initial debugging of the methods and for testing flights preceding each practical deployment of the system.

For the method description, let us denote the UAV position in the cartesian coordinates as $p_j(k) := \{x_j(k), y_j(k), z_j(k)\}$, its heading $\theta_j(k)$, and the camera heading $\varphi_j(k)$, where $k \in \{1, ..., N\}$ denotes indexes of transition points in the MPC control horizon. For MPC techniques, let us use control inputs $U_j(k) := \{v_j(k), w_j(k), K_j(k)\}, k \in \{1, ..., N\}$, where $v_j(k)$ is forward velocity, $w_j(k)$ ascent velocity, and $K_j(k)$ curvature of the trajectory, for control of the j-th UAV and angular velocity $\omega_j(k)$ for control of its gimbal with the camera. These control inputs are constant in-between of the transition points and lead the UAV from one transition point into a following one applying the model from [21]. This notation is used for the leader as well as the followers in the formation, such that $j \in \{L, 1, ..., n_F\}$.

A. Trajectory tracking of the leader with camera

As mentioned, two control problems are solved independently (control of UAV states $\psi_L(k) := \{p_L(k), \theta_L(k)\}, k \in \{1, ..., N\}$, and camera headings $\varphi_L(k), k \in \{1, ..., N\}$) to reduce the overall computational complexity in the proposed method.

Using the MPC methodology, we can formulate the first problem as an optimization of a vector $\Omega_{L,1} = [U_L(k)] \in \mathbb{R}^{3N}, k \in \{1, ..., N\}$. The objective function $J_{L,1}$, which is minimised and subject to a set of inequality constraints, is designed as

$$
\begin{aligned}
J_{L,1}(\Omega_{L,1}) = {} & \frac{\alpha}{N}\sum_{k=1}^{N} ||p_L^{ex}(k) - p_L(k)||^2 \\
& + \sum_{k=1}^{N} \frac{\beta_1}{N} K_L^{diff}(k)^2 + \frac{\beta_2}{N} v_L^{diff}(k)^2 + \frac{\beta_3}{N} w_L^{diff}(k)^2 \\
& + \frac{\gamma}{N}\sum_{k=1}^{N}\left(\min\left\{0, k\frac{\min\limits_{j\in\{1\cdots n_F\}} dist(p_L(k), \Omega^{\star}_{j,1}) - r_s}{\min\limits_{j\in\{1\cdots n_F\}} dist(p_L(k), \Omega^{\star}_{j,1}) - r_a}\right\}\right)^2 \\
& + \frac{\delta}{N}\sum_{k=1}^{N}\left(\min\left\{0, k\frac{dist(p_L(k), O) - r_s}{dist(p_L(k), O) - r_a}\right\}\right)^2 \\
& + \frac{\varepsilon}{N}\sum_{k=1}^{N}\left(\min\left\{0, ||p_{OoI}^{ex}(p_L^{ex}(k))) - p_L^{ex}(k)|| \right.\right. \\
& \qquad \left.\left. - ||p_{OoI}^{ex}(p_L^{ex}(k))) - p_L(k)||\right\}\right)^2 .
\end{aligned}
\tag{1}
$$

The first component of the cost function penalises deviations of the leader's position $p_L(k)$ along the planning interval N from the desired position on the path given by the experts $p_L^{ex}(k)$ as one of the inputs of the method. The second component is employed to suppress big deviations in control inputs (and so too aggressive behaviour of the system), where the notation $(\cdot)^{diff}(k) := (\cdot)(k-1) - (\cdot)(k)$ represents the difference of the particular control parameter from its previous value. The control inputs at $k = 0$ represent values that are implemented by controllers at the time of initialization of the optimization process. The third term ensures avoidance of other robots in the formation that are considered as dynamic obstacles for the leader. Function $dist(p_L(k)\Omega^{\star}_{j,1})$ returns the shortest distance between the position of the leader being optimized and the plan of the j-th follower obtained in the last call of the optimization process described in section III-B (let us denote results of the optimization processes by the notation $(\cdot)^{\star}$). Constant r_s is radius of a safety area around the robot in which the obstacles are considered in the avoidance function and r_a is radius of an avoidance area. The value of the third term approaches infinity if distance to an obstacle equal to r_a is reached. The fourth component, which protects the leader from collisions with static obstacles, uses the same avoidance function. Also the influence of this term is increased for an obstacles detected at the end of the control horizon (based on index k of the position $p_L(k)$ that contributes to the cost function), which ensures the obstacle avoidance functionality and protects the system from oscillations in a close proximity to obstacles. Function $dist(p_L(k), O)$ returns the shortest distance between the position of the leader and the obstacles. The fifth term is important due to the filming task as it penalises trajectories that are in a bigger distance to the position of the object of interest $p_{OoI}^{ex}(p_L^{ex}(k))$ in case of a deviation of the leader from the desired path by an obstacle. Avoiding an obstacle in a free space closer to the OoI, which is selected by the experts as the one that has to be observed by the follower at position $p_L^{ex}(k)$, decreases probability that the obstacle appears in the camera field of view.

The values of coefficients α, β_1, β_2, β_3, γ, δ, and ε in equation (1) influence behaviour of the system between often antagonistic requirements. For example with increasing values of $\beta_{(\cdot)}$, a smoother and less aggressive flight performance will be provided, while with increasing values of γ and δ the obstacle avoidance ability of the system is preferred.

The inequality constraints $r_a - dist(p_L(k), O) \leq 0, k \in \{1, ..., N\}$, and $r_a - \min\limits_{j}\{dist(\Omega_{L,1}, \Omega^{\star}_{j,1})\} \leq 0, j \in \{1, ..., n_F\}$, which have to be ensured in the optimization, support the avoidance function to ensure that a plan of the leader does not touch an obstacle by its avoidance region. In addition, control inputs in the optimization vector are limited by the following inequality constraints $v_L(k) - v_{L,max} \leq 0$, $v_{L,min} - v_L(k) \leq 0$, $w_L(k) - w_{L,max} \leq 0$, $w_{L,min} - w_L(k) \leq 0$, $K_L(k) - K_{L,max} \leq 0$, $K_{L,min} - K_L(k) \leq 0$, where $k \in \{1, ..., N\}$, including upper and lower bounds of the UAV controllers.

The second optimization problem is employed to obtain a smooth and feasible control of the camera heading (instructions for the gimbal or UAV controller in the case of the fixed mounting). Let us define the optimization vector in this

This is the author's version of an article that has been published. Changes were made to this version by the publisher prior to publication.
The final version of paper is available at https://doi.org/10.1109/ETFA.2017.8247654

problem as $\Omega_{L,2} = [\omega_L(k)] \in \mathbb{R}^N$, $k \in \{1, ..., N\}$, and the cost function, which is again minimised in the optimization, as

$$J_{L,2}(\Omega_{L,2}) = \zeta \sum_{k=1}^{N} diff_{ang}(\varphi_L^{ex}(k), \varphi_L(k))^2 + \eta \sum_{k=1}^{N} \omega_L^{diff}(k)^2, \quad (2)$$

where $\varphi_L^{ex}(k)$ is the desired camera heading at k-th transition point. The heading is obtained from the position of the object of interest $p_{OoI}^{ex}(p_L^{ex}(k))$ that has to be filmed/scanned at this moment, and the position of the leader at k-th transition point of the plan $\Omega_{L,1}^{\star}$ obtained as a solution of the first optimization problem. Although, the heading $\varphi_L^{ex}(k)$ is not explicitly given by the experts, we used the $(\cdot)^{ex}$ notation here, since they provide the desired position of the leader that should be as close as possible to the planned one and also the position of the OoI. The desired heading $\varphi_L^{ex}(k)$ should follow their intention as the orientation of the camera is naturally one of the most important factors influencing the required result in both filming techniques being solved in this paper.

The function $diff_{ang}(\psi_1, \psi_2)$ is defined for $\psi_1, \psi_2 \in \langle 0, 2\pi)$ as

$$diff_{ang}(\psi_1, \psi_2) = \begin{cases} \psi_1 - \psi_2 & if \quad \psi_1 - \psi_2 \leq \pi, \\ 2\pi - (\psi_1 - \psi_2) & if \quad \psi_1 - \psi_2 > \pi. \end{cases} \quad (3)$$

The constants ζ and η set influence of the objective trying to follow the desired camera orientation (the first term of the multi-objective function in eq. (2)) and the objective achieving not aggressive change of the orientation (the second term), respectively. The only constrain functions that have to be satisfied for optimization of the orientation are used to limit the UAV angular velocity as $\omega_L(k) - \omega_{L,max} \leq 0$ and $\omega_{L,min} - \omega_L(k) \leq 0$, $\forall k \in \{1, ..., N\}$. Although the camera heading control could be done by any simpler controller combined together with an interpolation of desired heading values, we rely on MPC to be able to synchronise it with the position control under the recoding horizon.

B. Trajectory tracking for followers with the light source

The desired positions of the n_F followers $p_{j,d}(k)$, $j \in \{1, ..., n_F\}$, $k \in \{1, ..., N\}$, in the formation are obtained from the positions of the leader (derived from its actual plan - the last result $\Omega_{L,1}^{\star}$ of the MPC position control) and the position of the currently filmed object of interest. The direction of the source of light (from the position of the follower to the OoI) and the optical axis of the camera have to form predefined angles $\pm 45^{\circ}$ in case of the *Three point lighting* approach and $\pm 90^{\circ}$ for the *Strong side-lightening* technique. The desired distance of the light is given by the requirements on the intensity of illumination. Naturally, such defined desired trajectory for the followers is not feasible for the UAVs (mainly when the leader suddenly switches its attention between two objects of interest). Therefore, the trajectory cannot be directly used as a desired control equilibrium, but its following needs to be integrated into the MPC framework taking into account motion and formation driving constraints applied in the optimization.

For followers control, we again assume that the orientation of the light source can be controlled independently to the UAV position control and so both these problems will be tackled in two separate optimization processes. In the trajectory tracking of the j-th follower, the optimization vector $\Omega_{j,1} = [v_j(k), w_j(k), K_j(k)] \in \mathbb{R}^{3N}, k \in \{1, ..., N\}$, collecting control inputs over the control horizon with N transition points is obtained by minimizing the multi-objective function

$$\begin{aligned} J_{j,1}(\Omega_{j,1}) = {} & \frac{\chi}{N} \sum_{k=1}^{N} ||p_{j,d}(k) - p_j(k)||^2 \\ & + \sum_{k=1}^{N} \frac{\iota_1}{N} K_j^{diff}(k)^2 + \frac{\iota_2}{N} v_j^{diff}(k)^2 + \frac{\iota_3}{N} w_j^{diff}(k)^2 \\ & + \frac{\kappa}{N} \sum_{k=1}^{N} \left(\min \left\{ 0, k \frac{\min\limits_{neigh_j} dist(p_j(k), \Omega_{neigh_j,1}^{\star}) - r_s}{\min\limits_{neigh_j} dist(p_j(k), \Omega_{neigh_j,1}^{\star}) - r_a} \right\} \right)^2 \\ & + \frac{\lambda}{N} \sum_{k=1}^{N} \left(\min \left\{ 0, k \frac{dist(p_j(k), O) - r_s}{dist(p_j(k), O) - r_a} \right\} \right)^2 \\ & + \frac{\mu}{N} \sum_{k=1}^{N} (\min \{0, \\ & \quad k \frac{||p_b(p_j(k), p_L^{\star}(k), \varphi_L^{\star}(k)) - p_j(k)|| - d_s}{||p_b(p_j(k), p_L^{\star}(k), \varphi_L^{\star}(k)) - p_j(k)|| - d_a} \right\} \Big)^2 . \end{aligned} \quad (4)$$

The first term in the objective function penalises deviation from the desired positions $p_{j,d}(k)$ of the j-th follower. The desired position depends on the position of the leader in the k-th transition point using its actual plan $\Omega_{L,1}^{\star}$, position of the object of interest that is filmed by the leader in the k-th transition point, and the requirements of the *Three point lighting* or *Strong side-lightening* approaches. The second term penalises too aggressive control behaviour as in the case of the leader's control. The third part of the cost function represents the mutual avoidance function, where set $neigh_j := \{L, 1 \cdots n_F\} \backslash j$ collects indexes of all remaining robots in the formation, which are the neighbours of the j-th follower. The fourth part ensures collision avoidance as for the leader. The last term, which is aimed to avoid having the UAV "illuminator" in the view of the camera, penalises the solutions in which the follower gets closer to the border of a pyramid representing space currently observed by the camera. The apex of the pyramid is roughly defined by the expected position of the leader $p_L^{\star}(k)$ following its actual plan $\Omega_{L,1}^{\star}$. Orientation of the pyramid corresponds with the planned orientation of the camera $\varphi_L^{\star}(k)$ according to $\Omega_{L,2}^{\star}$. In the cost function, distances between the position of the follower $p_j(k)$ and position $p_b(p_j(k), p_L^{\star}(k), \varphi_L^{\star}(k))$, which is the closest point on the pyramid (defined by $p_L^{\star}(k)$ and $\varphi_L^{\star}(k)$) from the position $p_j(k)$, smaller than a safety distance d_s are

This is the author's version of an article that has been published. Changes were made to this version by the publisher prior to publication.
The final version of paper is available at https://doi.org/10.1109/ETFA.2017.8247654

penalised. While d_s denotes the minimum desired distance from the border of the pyramid that is considered as safe, distance d_a is infeasible. The values of coefficients χ, ι_1, ι_2, ι_3, κ, λ, and μ set system behaviour similarly as in the case of the leader.

In the optimization process, the same inequality constraints protecting collisions with obstacles and other team members as in the position control of the leader are applied. The limits of control inputs for the followers may differ from the bounds used for the leader control, but the same structure of inequality constraints is used. Similarly, also the last term of the cost function (4) is supported by the inequality constraints $d_a - ||p_b(p_j(k), p^\star_L(k), \varphi^\star_L(k)) - p_j(k)|| \leq 0$, $\forall k \in \{1, ..., N\}$.

As mentioned, movement of the light source on a gimbal or change of UAV heading in case of a firm attachment of the light source is limited by motion constraints. Moreover, mainly for the *Three point lighting* approach, smooth changes of direction of the light source carried by the followers are required. We have to solve this problem in each control step under the MPC framework for each follower j and therefore we formalize it as an optimization problem with vector $\Omega_{j,2} = [\omega_j(k)] \in \mathbb{R}^N$, where $k \in \{1, ..., N\}$. The cost function that is minimised in the optimization is defined as

$$J_{j,2}(\Omega_{j,2}) = \nu \sum_{k=1}^{N} diff_{ang}(\varphi_{j,d}(k), \varphi_j(k))^2 + \xi \sum_{k=1}^{N} \omega_j^{diff}(k)^2, \quad (5)$$

where $\varphi_{j,d}(k)$ is desired heading of the light source at the k-th transition point obtained from the given position of the object of interest and the position of the j-th follower at the k-th transition point of the plan $\Omega^\star_{j,1}$, which is obtained as a solution of the first optimization problem. In comparison with the cost function used for the MPC control of the camera heading in eq. (2), here the desired orientation of the light is not given by the experts directly, but it is computed based on results of subsequently realized MPC steps for the leader and the followers composed into a compact formation via the rules of the *Three point lighting* and *Strong side-lightening* techniques. Nevertheless, the motivation of both parts of the cost function, meaning of the constants ν and ξ, and also the employed inequality constraints are the same as for control of the camera direction.

IV. Experimental verification and system deployment

The designed formation flying and stabilization system has been developed and verified in two robotic simulators, and the principles of lightening by a pair of UAVs were employed in two historical buildings including two large churches with a cooperation of historians, restorers, and filmmakers. The output of the system deployment was a set of images used by restorers for a plan of restoration works and two professionally edited documents broad-coasted by the main Czech TV and several other news providers (see the list at http://mrs.felk.cvut.cz/projects/cesnet). The parameters have been used as $n = 1$, $N = 8$, $\Delta t = 0.5$, $\alpha = 1.5$, $\beta_1 = 0.2$, $\beta_2 = 0.2$, $\beta_3 = 0.2$, $\gamma = 0.2$, $\delta = 0.01$, $\varepsilon = 0.1$, $\zeta = 0.1$, $\eta = 1$, $\chi = 1.5$, $\iota_1 = 0.5$, $\iota_2 = 0.5$, $\iota_3 = 0.1$, $\kappa = 0.1$, $\lambda = 0.1$, $\mu = 0.1$, $\nu = 0.1$, and $\xi = 1$ in all experiments.

The first simulator, V-Rep, was used for simulation of both lightening techniques implemented by the UAV formation since it provides a useful tool for visualization of shadows on 3D objects caused by onboard lights (see Fig. 5 for an example of the formation flying around a 3D statue). The second simulator, Gazebo, enables a very realistic verification of the formation stabilization and control approach. In this simulator, we have implemented a plugin that emulates a firmware of the Pixhawk low-level stabilization, which is used in our platform being primarily designed for the MBZIRC competition in Abu Dhabi (http://mrs.felk.cvut.cz/projects/mbzirc) together with Vijay Kumar Lab, University of Pennsylvania, http://kumarrobotics.org/ (for the HW system see [26]).

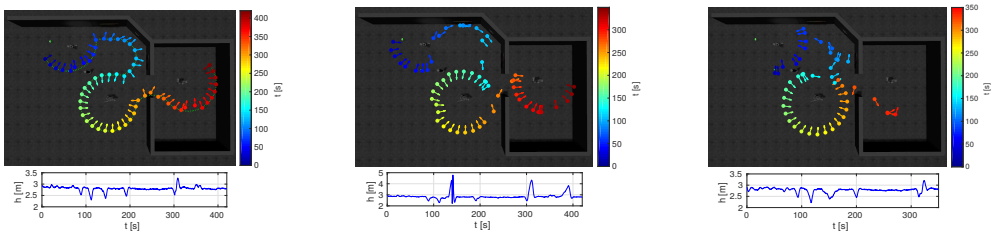
The simulator Gazebo enables to test the method even in more challenging conditions than are in historical buildings. For example in Fig. 2, a scenario inspired by an indoor space of Saint Nicholas Church in Prague was built to evaluate the system prior its using there. The testing scene includes six objects of interests and fourteen obstacles (including walls), which is much more complicated work-space in comparison with the tasks solved in the Saint Nicholas Church. The scenario in Fig. 2 with snapshots from one of the simulations in Fig. 4 verifies smooth transition between several consequent objects of interest (e.g. Fig. 4 (a-c)), the obstacle avoidance function (Fig. 4 (d-f)) and avoidance of collisions with other robots in the formation (Fig. 4 (g-h)). For the obstacle avoidance verification, the initial path was designed purposely infeasible for the formation by adding additional obstacles too close to the path and a failure of one of the follower was simulated (the follower with fill light, the robot on the right side of the formation, was suddenly stopped and the other UAVs were forced to avoid it).

The real deployment of the system in the Saint Nicholas Church located on the Old Town Square in Prague http://www.svmikulas.cz/en/ is shown in Fig. 6 and in document at https://youtu.be/g1NuPnLCFTg. The picture show formation stabilization aimed to film the statue of Jesus Christ and the pulpit providing illumination by the *Three point lighting* technique. The second deployment of the proposed UAV formation lightening approach in the Virgin Mary Church in Sternberk, Moravia, is presented in Fig. 1 and in video at https://youtu.be/-sTUwzFf_Mk.

V. Conclusion

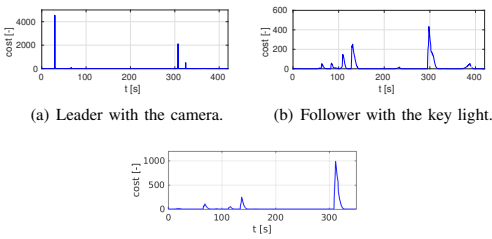
In this paper, a system designed for visual documentation and inspection of interiors of historical buildings by a formation of cooperating UAVs was described. The proposed solution of this application arises from our long-term basic research work in the field of UAV formation flying and it is mature enough to provide a solution that is already in use by historians and restorers to complement their work

This is the author's version of an article that has been published. Changes were made to this version by the publisher prior to publication.
The final version of paper is available at https://doi.org/10.1109/ETFA.2017.8247654



(a) Leader with the camera. (b) Follower with the key light. (c) Follower with the fill light.

Fig. 2. Trajectories of all formation members in the experiment in the complex environment. The arrows denote the heading of the camera in different time, and the green dotted line shows the desired trajectory. In the graph, h is the height which corresponds with the z coordinate of the leader.



(a) Leader with the camera. (b) Follower with the key light.

(c) Follower with the fill light.

Fig. 3. The progress of values of the cost function used for trajectory tracking.

in inaccessible places of large churches. From this point of view, the basic functionalities of the system achieved TRL 8 (the system has been tested and launched in real operations), while the full-scale system operability is currently in TRL 7-8 levels. The complete technology including obstacle avoidance and failure recovery has been developed and demonstrated in realistic robotic simulations.

Acknowledgments

The work has been supported by the Khalifa University in Abu Dhabi via the sponsorship for participation in the MBZIRC challenge, by the CESNET project for UAV mapping of historical locations, and by the Czech Science Foundation (GACR) under research project No. 17-16900Y.

References

[1] Y. Zhang and K.-L. Ma, "Lighting design for globally illuminated volume rendering," *IEEE Transactions on Visualization and Computer Graphics*, vol. 19, no. 12, pp. 2946–2955, 2013.

[2] J. Birn, *Digital Lighting and Rendering, 2013*. New Riders, year=, ch. Three-Point Lighting for 3D Renderings.

[3] T. Baca, G. Loianno, and M. Saska, "Embedded model predictive control of unmanned micro aerial vehicles," in *MMAR*, 2016.

[4] J. Faigl, T. Krajnik, J. Chudoba, L. Preucil, and M. Saska, "Low-Cost Embedded System for Relative Localization in Robotic Swarms," in *IEEE ICRA*, 2013.

[5] T. Krajnik, M. Nitsche, J. Faigl, P. Vanek, M. Saska, L. Preucil, T. Duckett, and M. Mejail, "A practical multirobot localization system," *Journal of Intelligent & Robotic Systems*, vol. 76, no. 3-4, pp. 539–562, 2014.

[6] W. Sheng, A. Okamoto, and S. Tanaka, "Visual point-based analysis of laser-scanned historical structures," in *International Conference on Culture and Computing*, 2015.

[7] P. K. Allen, I. Stamos, A. Troccoli, B. Smith, M. Leordeanu, and Y. C. Hsu, "3d modeling of historic sites using range and image data," in *IEEE ICRA*, 2003.

[8] N.-J. Shih, H.-J. Wang, C.-Y. Lin, and C.-Y. Liau, "3d scan for the digital preservation of a historical temple in taiwan," *Advances in Engineering Software*, vol. 38, no. 7, pp. 501 – 512, 2007.

[9] R. Zlot, M. Bosse, K. Greenop, Z. Jarzab, E. Juckes, and J. Roberts, "Efficiently capturing large, complex cultural heritage sites with a handheld mobile 3d laser mapping system," *Journal of Cultural Heritage*, vol. 15, no. 6, pp. 670 – 678, 2014.

[10] P. S. Blaer and P. K. Allen, "View planning and automated data acquisition for three-dimensional modeling of complex sites," *Journal of Field Robotics*, vol. 26, no. 11-12, pp. 865–891, 2009.

[11] J. Fernandez-Hernandez, D. Gonzalez-Aguilera, P. Rodriguez-Gonzálvez, and J. Mancera-Taboada, "Image-based modelling from unmanned aerial vehicle (uav) photogrammetry: An effective, low-cost tool for archaeological applications," *Archaeometry*, vol. 57, no. 1, pp. 128–145, 2015.

[12] N. Hallermann, G. Morgenthal, and V. Rodehorst, "Vision-based monitoring of heritage monuments: Unmanned aerial systems (uas) for detailed inspection and high-accuracy survey of structures," *WIT Transactions on The Built Environment*, vol. 153, no. 12, p. 621, 2015.

[13] K. Themistocleous, M. Ioannides, A. Agapiou, and D. G. Hadjimitsis, "The methodology of documenting cultural heritage sites using photogrammetry, uav, and 3d printing techniques: the case study of asinou church in cyprus," 2015.

[14] M. Hess, V. Petrovic, D. Meyer, D. Rissolo, and F. Kuester, "Fusion of multimodal three-dimensional data for comprehensive digital documentation of cultural heritage sites," in *Digital Heritage*, 2015.

[15] C. Teuliere, E. Marchand, and L. Eck, "3-d model-based tracking for uav indoor localization," *IEEE Transactions on Cybernetics*, vol. 45, no. 5, pp. 869–879, 2015.

[16] S. Minaeian, J. Liu, and Y. J. Son, "Vision-based target detection and localization via a team of cooperative uav and ugvs," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 46, no. 7, pp. 1005–1016, 2016.

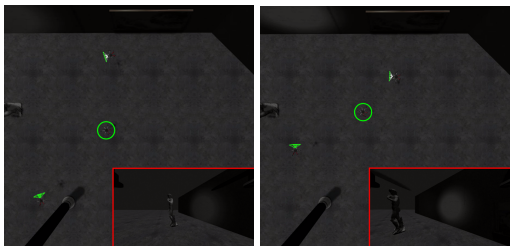
[17] A. L. Majdik, L. Tizedes, M. Bartus, and T. Szirányi, "Photogrammetric 3d reconstruction of the old slaughterhouse in budapest," in *IWCIM*, 2016.

[18] M. Saska, V. Spurny, and V. Vonasek, "Predictive control and stabilization of nonholonomic formations with integrated spline-path planning," *Robotics and Autonomous Systems*, vol. 75, no. Part B, pp. 379–397, 2016.

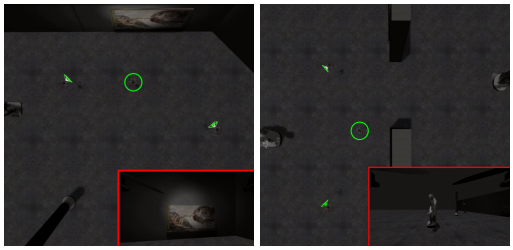
[19] M. Saska, J. Mejia, D. Stipanovic, V. Vonasek, K. Schilling, and L. Preucil, "Control and Navigation in Manoeuvres of Formations of Unmanned Mobile Vehicles," *European Journal of Control*, vol. 19, no. 2, pp. 157–171, 2013.

[20] M. Saska, J. S. Mejia, D. M. Stipanovic, and K. Schilling, "Control and navigation of formations of car-like robots on a receding horizon," in *Proc of 3rd IEEE Multi-conference on Systems and Control*, 2009.

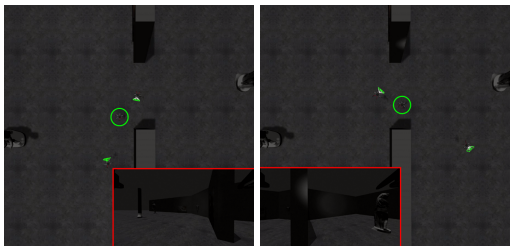
This is the author's version of an article that has been published. Changes were made to this version by the publisher prior to publication.
The final version of paper is available at https://doi.org/10.1109/ETFA.2017.8247654



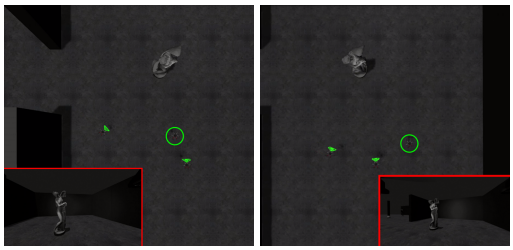
(a) Time - 65s. (b) Time - 70s.



(c) Time - 80s. (d) Time - 305s.

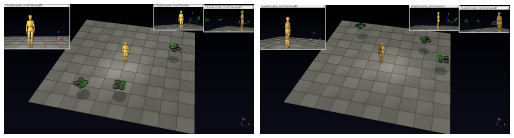


(e) Time - 312s. (f) Time - 320s.



(g) Time - 368s. (h) Time - 380s.

Fig. 4. Snapshots of the formation during the transition to another object of interest (a-c), during the transition to another object of interest and concurrent avoiding the obstacle (d-f), and during avoiding the broken follower carrying the fill light (g-h). The three UAVs are flying in a formation to realize a film crew consisting of a cameraman (the robot denoted by the green circle) and two illuminators (the robots with pyramids representing the lights) realizing the *Three point lighting* approach in an environment inspired by the interior of the Saint Nicholas church.



(a) (b)

Fig. 5. Simulation of the *Three point lighting* technique in V-Rep.

(a) Filming a statue of Jesus Christ. (b) Filming a pulpit.

Fig. 6. A hovering formation performing part of the *Three point lighting* technique in the Saint Nicholas church.

[21] M. Saska, V. Vonasek, T. Krajnik, and L. Preucil, "Coordination and Navigation of Heterogeneous MAV&UGV Formations Localized by a 'hawk-eye'-like Approach Under a Model Predictive Control Scheme," *International Journal of Robotics Research*, vol. 33, no. 10, pp. 1393–1412, 2014.

[22] M. Saska, Z. Kasl, and L. Preucil, "Motion Planning and Control of Formations of Micro Aerial Vehicles," in *The 19th IFAC World Congress*, 2014.

[23] M. Saska, V. Vonásek, T. Krajník, and L. Přeučil, "Coordination and navigation of heterogeneous uavs-ugvs teams localized by a hawk-eye approach." in *IEEE/RSJ IROS*, 2012.

[24] V. Spurny, T. Baca, and M. Saska, "Complex manoeuvres of heterogeneous mav-ugv formations using a model predictive control," in *MMAR*, 2016.

[25] M. Saska, V. Spurny, and L. Preucil, "Trajectory Planning and Stabilization for Formations Acting in Dynamic Environments," in *Progress in Artificial Intelligence*, 2013.

[26] M. Saska, T. Baca, J. Thomas, J. Chudoba, L. Preucil, T. Krajnik, J. Faigl, G. Loianno, and V. Kumar, "System for deployment of groups of unmanned micro aerial vehicles in GPS-denied environments using onboard visual relative localization," *Autonomous Robots. First online.*, 2016.

Chapter 8

Autonomous Reflectance Transformation Imaging by a Team of Unmanned Aerial Vehicles

This section presents the last core publication of this compiled Ph.D. thesis called *Autonomous Reflectance Transformation Imaging by a Team of Unmanned Aerial Vehicles* [4c], published in Q1 journal IEEE Robotics and Automation Letters in 2020. This publication presents a novel approach proposed for the realization of a Reflectance Transformation Imaging technique (RTI) by a team of UAVs.

[4c] V. Kratky, P. Petracek, **V. Spurny**, *et al.*, "Autonomous Reflectance Transformation Imaging by a Team of Unmanned Aerial Vehicles," *IEEE Robotics and Automation Letters*, vol. 5, no. 2, pp. 2302–2309, 2020

The **introductory** section of the paper explains our motivation for developing an autonomous system enabling the RTI. The main advantage of the proposed solution compared to already existing methods is the ability to perform the RTI scanning procedure in places that are difficult or impossible to access for humans. This section also lists related works focusing on scanning the interiors of buildings, which is one of the target applications of this fundamental research.

The main contribution of this core publication is explained in the section **distant autonomous RTI method**. This section contains three approaches to the generation of sequences of RTI positions and an MPC-based framework able to perform collision-free flight between generated positions and heading control of the light. These three various approaches to the generation of sequences of RTI positions are compared regarding time requirements and the length of the resulting path in the **experimental results** section of the paper. However, only one of the proposed methods was approved by representatives of the heritage institute for deployment in historical objects. This solution includes a novel approach to the generation of human-predictable trajectories for simple monitoring of correct behavior of particular UAVs by safety pilots, while also preserving an effort to generate short trajectories. This property significantly increases the safeness of the entire solution, which is highly important to UAV deployment in historical buildings due to the costs of scanned artifacts. The functionality of the complete proposed system had been verified through several experiments in the realistic Gazebo robotic simulator and within real-world experiments deploying two autonomous UAVs in the interior of the Church of St. Mary Magdalene in Chlumín, CZE.

The manuscript summarizes the fundamental research done by V. Kratky (the main author) for his diploma thesis, which was supervised by the author of this Ph.D. thesis who also significantly contributed to the initial design of the presented approach.

This is the author's version of an article that has been published. Changes were made to this version by the publisher prior to publication.
The final version of paper is available at https://doi.org/10.1109/LRA.2020.2970646

IEEE ROBOTICS AND AUTOMATION LETTERS. PREPRINT VERSION. ACCEPTED JANUARY, 2020 1

Autonomous Reflectance Transformation Imaging by a Team of Unmanned Aerial Vehicles

Vít Krátký, Pavel Petráček, Vojtěch Spurný, and Martin Saska

***Abstract*—A Reflectance Transformation Imaging technique (RTI) realized by multi-rotor Unmanned Aerial Vehicles (UAVs) with a focus on deployment in difficult to access buildings is presented in this paper. RTI is a computational photographic method that captures a surface shape and color of a subject and enables its interactive re-lighting from any direction in a software viewer, revealing details that are not visible with the naked eye. The input of RTI is a set of images captured by a static camera, each one under illumination from a different known direction. We present an innovative approach applying two multi-rotor UAVs to perform this scanning procedure in locations that are hardly accessible or even inaccessible for people. The proposed system is designed for its safe deployment within real-world scenarios in historical buildings with priceless historical value.**

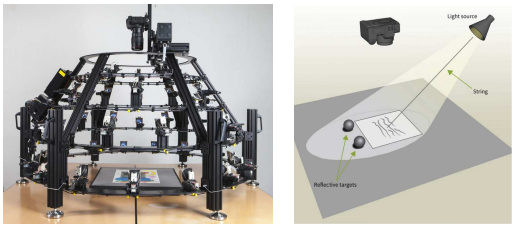
***Index Terms*—Aerial Systems: Applications, Cooperating Robots, Multi-Robot Systems**

I. INTRODUCTION

REFLECTANCE Transformation Imaging (RTI) is an image-based rendering method widely used by experts in the field of archaeology, restoration and historical science [1]–[7]. Based on the set of images with varying known lighting, a representation of an image is produced by RTI, that enables to view a captured object lit from an arbitrary direction and therefore to easily inspect the three-dimensional character of the object without the need to capture thousands of photographs with lighting from all possible directions.

The most traditional approaches for gathering the desired set of images is an RTI dome (see Fig. 1a) and the Highlight RTI method [8]. The RTI dome includes tens of light-emitting diodes (LEDs) placed on the inner surface of a hemisphere and a camera placed on its top. During the image capturing phase, the RTI dome is placed above the scanned object, and the LEDs are sequentially lit up while the camera is capturing images. Each image is then labelled with the corresponding lighting vector computed from the known position of particular LEDs.

Using the Highlight RTI method (H-RTI), a source of light is manually placed at unknown positions in a constant distance from the scanned object, while a camera mounted on the tripod is capturing images. Respective lighting vectors are then computed from the reflection detected on the high reflective object (metal ball) placed next to the scanned artifact. An example of a setup for the H-RTI method is illustrated in Fig. 1b.



(a) RTI dome (b) highlight RTI

Fig. 1: Illustration of traditional approaches to the realization of RTI method. Image sources: https://www.idigbio.org/wiki/images/7/70/Graham_0429.pdf and https://historicengland.org.uk/images-books/publications/multi-light-imaging-heritage-applications/heag069-multi-light-imaging/

The drawback of both methods is that the scanned object has to be directly accessible to humans, which is difficult to achieve in large historical and sacred buildings. Thus it significantly limits the usage of this very powerful technique. We propose to solve this problem by applying two cooperating multi-rotor UAVs equipped with a camera and light source. The UAV team is able to gather the set of images with corresponding lighting vectors of objects located at places hardly accessible or even inaccessible for people, and much faster than using the H-RTI approach. During the scanning process, the UAV carrying a camera is hovering steadily in the air, while the UAV equipped with a light source is flying around to provide the lighting from different directions.

Although the scanning process could be performed manually, regardless of the experience of operators, the manual navigation of UAVs to desired positions is typically less precise than in the case of autonomous control. Moreover, the desired scanning locations are assumed to be located high above the ground and hence far from the operators, which increases the difficulty and danger of manual control of two UAVs flying close to each other. Therefore, the proposed method relies on using two fully autonomous and self-localized UAVs. Although we present the system as autonomous, each UAV is supposed to be monitored by an operator, who is prepared to take over the control in case of an unexpected behavior. This requirement is given by the aviation authority for flying outdoors (e.g., for scanning statues, mosaics and plasters on the exteriors of churches and castles - see video https://youtu.be/lTRqd1gQOAI) and by the

Manuscript received: September 10, 2019; Revised December 19, 2019; Accepted January 15, 2020.

This paper was recommended for publication by Editor Jonathan Roberts upon evaluation of the Associate Editor and Reviewers' comments. This work was supported by project no. DG18P02OVV069 in program NAKI II, by CTU grant no. SGS17/187/OHK3/3T/13, and by the Grant Agency of the Czech Republic under grant no. 17-16900Y.

Authors are with the Faculty of Electrical Engineering, Czech Technical University in Prague, Technická 2, Prague 6, {kratkvit|petrapa6|spurnvoj|saskaml}@fel.cvut.cz.

Digital Object Identifier (DOI): see top of this page.

This is the author's version of an article that has been published. Changes were made to this version by the publisher prior to publication.
The final version of paper is available at https://doi.org/10.1109/LRA.2020.2970646

heritage institute for flying indoors (see Fig. 2 and videos from Saint Nicholas Church at Old Town Square in Prague (https://youtu.be/g1NuPnLCFTg), Grotto Gorzanow, Poland (https://youtu.be/6mRYxciDLCM), and St. Anne's and St. Jacob's Church in Stará Voda (https://youtu.be/yNc1WfebIag), where autonomous UAVs have been applied).

The presented application is specific due to cooperation with experts from the Czech National Heritage Institute (https://www.npu.cz/en), who have introduced requirements and constraints, which are untraditional from the robotic point of view. Therefore, two different approaches to the generation of lighting positions and determination of an ideal sequence of these positions to achieve sufficiently good coverage of lighting for the RTI technique are presented in this paper. One of them is based on the method using Fibonacci lattice [9] for achieving approximately equal distribution of points on the sphere, and it applies the approaches for the solution of the Traveling Salesman Problem (TSP) to find a path connecting these positions. The second proposed approach is aimed to find a compromise among the optimality, robotic constraints, and requirements of the aviation authorities and the heritage institute that require paths producing predictable and easy to follow movement of UAVs, which is optimal regarding the UAV deployment in historical buildings.

Fig. 2: Deployment of the proposed system for autonomous RTI by a team of UAVs in Church of St. Mary Magdalene in Chlumín. Multimedia material of the experiment is available at http://mrs.felk.cvut.cz/papers/rti2020ral.

A. State-of-the-art and contribution

Single manually controlled UAVs are being commercially used in numerous scenarios, both outdoor and indoor. Nevertheless, the number of possible applications can be significantly increased by introducing autonomous cooperative teams of UAVs, which is the aim of this paper. One of such applications is the documentation of interiors of historical buildings with distributed lighting, which is motivated by the preservation of cultural heritage in the form of digital documentation [10]. It provides the ability to perform later reconstructions of already destroyed historical buildings or art pieces, and also provides the ability to analyze this data and plan the future restoration work without repetitive direct access to particular artifacts.

The documentation of buildings is problematic due to its time complexity and limited accessibility by humans, which naturally leads to the introduction of semi-autonomous or autonomous systems developed for this purpose. Works related to the scanning of buildings are mostly interested in the planning of the best sensing locations [11]–[13] and only a few of them aim to exploit autonomous vehicles. In [14], an unmanned ground vehicle (UGV) equipped with a laser scanner capable of autonomous planning of scanning locations and moving through a large scale outdoor environment is introduced. In [15], authors exploit advantage of UAV systems to operate in larger space by applying them to autonomous inspection of industrial chimneys. Regarding the documentation of particular artifacts in interiors of historical buildings, the only work we have found is [16], where technology assisting an operator of a single UAV explicitly developed for this application is described.

Regarding the multi-robot systems works presenting systems deploying UGVs or UAVs for the documentation or mapping of buildings, the robotic groups are focused on reducing the overall mission time or on expanding the scanned area [17], [18], but the direct cooperation of robots is not exploited. The cooperative lighting by a UAV team introduced in [10], and extended for use of the RTI technique in this paper, is unique in comparison to all the aforementioned works since it employs a team of cooperating UAVs in tasks that cannot be solved by a single UAV only in principle. The proposed method in [10] is exceptional in its approach to actively influence its surrounding environment in order to increase quality and variety of gathered digital material.

In this work, we introduce the first system for autonomous realization of the RTI technique independent on the location of scanned objects, which takes the advantage of our previous works on formation control [10], [19]–[21].

II. PROBLEM DESCRIPTION

The problem of autonomous realization of Reflectance Transformation Imaging technique consists of 1) determining a set of desired positions of a light source, 2) finding a feasible trajectory so that a UAV can provide illumination from these positions, 3) precise mutual localization of UAVs, and 4) processing the captured images for computation of the desired representation of an image. The team of UAVs consists of one UAV equipped with a high-resolution camera and one UAV carrying a light source. Both UAVs are assumed to be capable of steady hovering in the air and controlling the orientation of the camera and the light independently of their motion.

We suppose that the UAVs operate in a known environment represented by a map, obtained from a three-dimensional scan of the historical building, and they are equipped with necessary sensors and software for their precise localization and state estimation [22]. The map of the environment is obtained from a three-dimensional terrestrial laser scanner, providing an incomplete map with missing data in occluded out-of-view locations. Such map is sufficient for localization, however does not provide sufficiently precise and complete models of particular artifacts. Requirements on the scanning process of an object are given by specification of the RTI technique [23] and a position of the object selected for scanning is known prior the mission. Both UAVs are able to accurately follow the trajectory given by the sequence of configurations in an available map of the environment [24].

The output of the system is the requested representation of the image computed from the set of images taken with the

This is the author's version of an article that has been published. Changes were made to this version by the publisher prior to publication.
The final version of paper is available at https://doi.org/10.1109/LRA.2020.2970646

camera carried by the UAV. Corresponding lighting vectors, which are needed for computation of this representation, are obtained from a known position of the scanned object and positions of the UAV carrying the light.

III. PRELIMINARIES

A. Reflectance Transformation Imaging

Reflectance Transformation Imaging (RTI) is an image-based rendering method used for obtaining a representation of an image that enables it to be displayed under arbitrary lighting conditions. One type of such representation is the Polynomial Texture Map (PTM), which was proposed by T. Malzbender [23]. In contrast to the common representation of an image, where each pixel has assigned three static values for red, green and blue color (RGB), the simplified version of PTM represents the intensity of each color channel $I_{c,x,y}$, $c \in \{red, green, blue\}$ of the pixel at position (x, y) by function

$$I_{c,x,y} = f(l_u, l_v), \tag{1}$$

where l_u and l_v are elements of lighting vector and the function $f(\cdot)$ is a second-order bi-quadratic polynomial function with varying coefficients $\alpha_{i,c}$ for particular pixels (x, y). Thus the intensity $I_{c,x,y}$ of each color can be interpreted as

$$\begin{aligned} I_{c,x,y} = \alpha_{1,c} l_u^2 + \alpha_{2,c} l_v^2 + \alpha_{3,c} l_u l_v + \alpha_{4,c} l_u + \\ + \alpha_{5,c} l_v + \alpha_{6,c}, \; c \in \{red, green, blue\}. \end{aligned} \tag{2}$$

The input of the RTI method is a set of images taken from the same viewpoint under varying known lighting conditions, where each image in the set has assigned corresponding lighting vector. With the use of this data, coefficients in equation (2) can be computed for all pixels and their color channels (see [23] for details).

B. Localization

Precise determination of position and orientation of UAVs is a crucial assumption for the good performance of the introduced documentation method. Since we aim at the deploying of the system mostly in indoor environments, we rely on the approach presented in [25], which is capable of working in environments without a sufficient signal from Global Navigation Satellite Systems (GNSS). The method requires UAVs to be equipped with one 360° laser scanner (such as a lightweight RP-Lidar), and two distance sensors (e.g., Garmin LIDAR-Lite v3) oriented downwards and upwards with respect to the frame of UAV. A combination of Iterative Closest Point (ICP) and particle filter algorithm is applied to find the position and orientation of the UAV relative to a three-dimensional point cloud of the environment obtained from a terrestrial laser scanner.

IV. DISTANT AUTONOMOUS RTI METHOD

Methods designed for the realization of the RTI scanning technique by a team of UAVs, which are described in the following sections, are highly influenced by the requirements of experts from the field of restoration and historical science, where key factors are safety and deployability independently to an external infrastructure.

A. Generation of the set of lighting positions

To achieve a good coverage of lighting to a general object during RTI scanning, the lighting vectors need be uniformly distributed over the range defined by the minimum and maximum lighting angles in horizontal ($\lambda_{h,min}$, $\lambda_{h,max}$) and vertical ($\lambda_{v,min}$, $\lambda_{v,max}$) direction. The intensity of lighting presented at the scanned object should be the same for all lighting directions.

Given these two requirements and assumption that the intensity of the light source is constant, we can determine that the desired positions of light sources are distributed on a cap of the sphere with its center located at the position of the scanned object. This task can be defined as the problem of uniform distribution of points on the sphere. Since this problem has an exact solution only for particular cases [26], we apply an approximate approach based on the Fibonacci lattice. Inputs of this process are the number of desired lighting positions to be uniformly distributed over the area defined by angles $\lambda_{d,m}$, $d \in \{h, v\}$, $m \in \{min, max\}$, position of an Object of Interest (OoI), and the desired lighting distance. The resulting set of points Λ_c computed within this process is constructed as

$$\Lambda_c = \Lambda \cup P_i, \tag{3}$$

where Λ is the set of desired lighting positions and $P_i \in \mathbb{R}^3$ is the initial position of the UAV carrying the light.

B. Determination of the optimal sequence

An optimal closed path connecting all the desired RTI lighting positions in the set Λ_c with respect to a certain criterion (minimum energy, shortest path, minimum time) needs to be found. This problem can be defined as TSP, which is usually solved by splitting it into two subproblems - finding paths between all possible pairs of positions from the set Λ_c and finding the optimal sequence of these paths with respect to a certain criterion. The final path is then given as a connection of paths in the optimal sequence. Using this approach, it is difficult to guarantee feasibility of a composed path with respect to constraints given by the kinematic model of a moving robot. Nevertheless, the RTI method requires a static illumination while capturing an image, and so the UAV carrying the light has to be static for taking each picture in the sequence. Therefore, the UAV should stop at every position from Λ_c and the problem of an unfeasible path in connections of curve segments does not need to be considered here.

Considering the expected application of the system, we propose to use the minimum energy as the optimization criterion for the solution of proposed alternative of TSP, which also leads to maximization of possible flight time. Based on our experiments, the energy consumption along the closed trajectory flown at constant velocity is proportional to the length of this trajectory and does not depend on the direction of flight. By combining the observations mentioned above and considering an obstacle-free environment, the problem of finding the optimal sequence of the lighting positions is completely defined as the Euclidean TSP (ETSP). For solution of this problem, we have applied the solver using

This is the author's version of an article that has been published. Changes were made to this version by the publisher prior to publication.
The final version of paper is available at https://doi.org/10.1109/LRA.2020.2970646

Lin-Kernighan heuristic [27] (LKH solver), which belongs to the most efficient approximate algorithms for solution of TSP. An example of path produced with the described approach (further referenced as Fib-LKH) is presented in Fig. 7c.

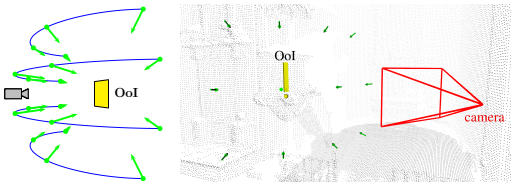
C. Safety pilot predictable approach (SPPA)

In this section, we present an alternative approach to the obtaining of scanning plan that aims at the generation of lighting positions close to uniform distribution and finding a short path connecting these positions while complying to requirements on human predictability of the resulting trajectory. Thus this method decreases the time needed for recognition of the faulty behavior by a safety pilot, who monitors the UAV during the scanning process. This approach is motivated by a technique used by restorers during the manual acquisition of images for RTI method in [28].

The proposed method for obtaining the set of desired lighting positions uses as inputs the border lighting angles $\lambda_{d,m}$, $d \in \{h, v\}$, $m \in \{min, max\}$, the position of scanned object $P_{OoI} \in \mathbb{R}^3$, the orientation of camera defined with yaw and pitch angle $(\psi_{cam}, \zeta_{cam})$, the desired distance between the light and scanned object d_l, and the desired number of samples of the lighting angles v_s in vertical direction for which holds $v_s \geq 2$. In the first step of the method, a set of samples of vertical lighting angles Λ_v from interval $\langle \lambda_{v,max}, \lambda_{v,min} \rangle$ is obtained so that they are equally distributed over this interval, $|\Lambda_v| = v_s$, $\min(\Lambda_v) = \lambda_{v,min}$ and $\max(\Lambda_v) = \lambda_{v,max}$. Subsequently one spline on which possible positions of the light lie is constructed for each sample λ_v from Λ_v. These splines are parts of a circle and with given $P_{OoI} = [x_{OoI}, y_{OoI}, z_{OoI}]^T$ are defined as

$$\begin{aligned} x_s &= x_{OoI} - d_l \cos(\lambda_v + \zeta_{cam}) \cos(\lambda_h + \psi_{cam}), \\ &\qquad \lambda_h \in \langle \lambda_{h,min}, \lambda_{h,max} \rangle \\ y_s &= y_{OoI} - d_l \cos(\lambda_v + \zeta_{cam}) \sin(\lambda_h + \psi_{cam}), \\ &\qquad \lambda_h \in \langle \lambda_{h,min}, \lambda_{h,max} \rangle \\ z_s &= z_{OoI} - d_l \tan(\lambda_v + \zeta_{cam}). \end{aligned} \tag{4}$$

The splines defined by equation (4) are graphically illustrated in Fig. 3a. The desired distance between the lighting positions



(a) illustration (b) experiment in real environment

Fig. 3: The example of the generated set of RTI goals marked with green dots and arrows. The yellow rectangle identifies the scanned object, and the blue curves indicate the horizontal splines, that represent possible positions of the RTI goals.

on one spline s_d is determined by the equation

$$s_d = \frac{d_l(\lambda_{v,max} - \lambda_{v,min})}{v_s}, \tag{5}$$

which corresponds to the shortest distance between two neighboring splines traveled on the surface of the spherical cap. The number of lighting positions on each spline is defined as

$$n_s(\lambda_v) = 1 + \left\lfloor \frac{d_l \cos(\lambda_v)(\lambda_{h,max} - \lambda_{h,min})}{s_d} \right\rceil. \tag{6}$$

The set of sample positions $\Lambda_h(\lambda_v)$ on spline corresponding to angle λ_v is obtained as

$$\Lambda_h(\lambda_v) = \left\{ \lambda_{h,min} + \frac{\lambda_{h,max} - \lambda_{h,min}}{2} \right\}, \tag{7}$$

if $n_s(\lambda_v) = 1$ and

$$\Lambda_h(\lambda_v) = \left\{ \lambda_{h,min} + k \frac{\lambda_{h,max} - \lambda_{h,min}}{n_s - 1} \middle| \; k \in \{0, 1, \ldots, n_s - 1\} \right\}, \tag{8}$$

if $n_s = (\lambda_v) \geq 2$. The complete set of the desired lighting positions Λ_c generated with the SPPA is defined as

$$\Lambda_c = \{\Lambda_h(\lambda_v) | \lambda_v \in \Lambda_v\} \cup P_i. \tag{9}$$

As the first step of the SPPA, the current position of the UAV carrying the light is added at the beginning of the ideal sequence of the RTI positions S_p. Then the closest pair of the RTI positions P_s, $P_e \in \mathbb{R}^3$ on the vertical boundaries needs to be found to select the higher one as the start point and the lower one as the end point among RTI positions. For P_s, P_e holds

$$\begin{aligned} P_s, P_e = \arg \min_{P_{i,j}, P_{k,l}} & \operatorname{dist}(P_{i,j}, P_i) + \operatorname{dist}(P_{k,l}, P_i), \\ s.t. \quad & i = k + 1, \\ & (j, l) \in \{(1, 1), (|\lambda_{h,i}|, |\lambda_{h,k}|)\}, \end{aligned} \tag{10}$$

where $P_{i,j}$ stands for the RTI position in the i-th row and j-th column, function $\operatorname{dist}(\cdot)$ returns the Euclidean distance between two positions given as arguments, and $\lambda_{h,i}$ stands for the set of RTI positions in the i-th row. The position P_s is then added to the sequence of positions S_p. After that, all positions on the vertical boundary on the way up to the highest row are added to S_p. By these three steps, one of the corner positions in the highest row is reached. In the following stages, the procedure depends on the number of rows.

In the case of an even number of horizontal rows, the RTI positions are added line by line with switching the left-right and the right-left direction, and omitting the points that lie on the same vertical boundary as P_s. After reaching the last admissible position in the most bottom row, the remaining points are added from the bottom row up to and including the P_e into S_p. Finally, the P_i is added at the end of the S_p to ensure the return to the initial position. The graphical illustration of this process is shown in Figures 4a, 4b and 4c.

The solution for an odd number of horizontal lines is derived from the solution for even number of rows with several modifications. Firstly, the pair of consequent horizontal lines (indicated by pair of indices $(h_{o,1}, h_{o,2})$) with minimum number of RTI positions is determined. Then, the solution for an odd number of rows is the same as in the case of

This is the author's version of an article that has been published. Changes were made to this version by the publisher prior to publication.
The final version of paper is available at https://doi.org/10.1109/LRA.2020.2970646

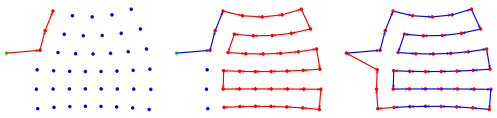


Fig. 4: Illustration of the procedure of determining the safety pilot predictable sequence of RTI positions for even number of horizontal rows. The green dot marks the initial state P_i, the blue dots stand for the unvisited RTI positions, and the red dots for already visited RTI positions. The arrows show the transitions between particular RTI positions, where the red arrows stand for the transitions added during the last step.

even number of rows until the procedure reaches the pair $(h_{o,1}, h_{o,2})$. The particular RTI positions within this pair of rows are traversed either in an up-side-down-side manner (see Fig. 7a for example), or by following the positions in row $h_{o,1}$ to the opposite side, then flying back to the starting side of row $h_{o,2}$ and again following this row to the opposite side. From there, the situation and also the solution is again the same as in the case of an even number of rows.

D. Trajectory Generation and Tracking

The desired trajectory for the RTI (Ω) is generated by the sampling of direct straight paths between consequent RTI positions with the sampling distance d_{RTI} (computed based on the desired velocity), without considering any obstacles. To achieve precise lighting conditions, the UAV carrying the light is supposed to hover at the desired position while taking a photo. This requirement is introduced into the presented system by multiple recurrences of the desired RTI position as the transition point of Ω after each fly-over to the next RTI position. The number of these repetitions is proportional to the time required for the stabilization.

To achieve a reliable following of the desired trajectory Ω in an environment with obstacles and in the presence of disturbances, which cannot be omitted in real systems, the trajectory tracking during the RTI procedure is defined as an optimization task within the MPC framework in the proposed system. Thanks to the independence of the position and orientation control in case of multi-rotor UAVs, the optimization loop can be divided into two separate tasks.

The position control is defined as a nonlinear constrained optimization task over a sequence of control inputs $\mathcal{U}_p(t)$ starting at time t with an objective function J_p, and set of nonlinear constraints $g_p(\cdot)$ on the horizon of length N as

$$\begin{aligned} \mathcal{U}_p(t)^* = \arg\min J_p(\mathcal{U}_p(t)), \\ s.t.\ g_p(\mathcal{U}_p(t), \mathcal{O}(t)) \quad \leq 0, \end{aligned} \tag{11}$$

where $\mathcal{O}(t)$ is the set of all obstacles present at time t in the environment, including the UAV carrying the camera.

The objective function $J_p(\cdot)$ is defined as the weighted sum

$$J_p = \alpha J_{pos} + \beta J_c + \gamma J_{obs} + \delta J_{rti}, \tag{12}$$

where J_{pos} stands for the part penalizing the deviations from the desired trajectory, J_c is the part penalizing the changes in sequence of control inputs, and J_{obs} responds for the penalization of trajectories in the proximity of obstacles. The value of J_p is increased by adding J_{rti} for trajectories that lead to occlusions caused by the UAV carrying light or lead to shades in the image caused by the lighting from behind the UAV carrying the camera. Coefficients $\alpha, \beta, \gamma,$ and δ are weights used for the scaling of particular parts of the objective function.

The function J_{rti}, which was proposed specifically for this application, is defined as

$$J_{rti} = \sum_{k=1}^{N} \left(\min \left\{ 0, \frac{d_{FoV}(k) - r_{d,FoV}}{d_{FoV}(k) - r_{a,FoV}} \right\} \right)^2, \tag{13}$$

where $r_{d,FoV}$ and $r_{a,FoV}$ are detection and avoidance radii with respect to camera Field of View (FoV), and $d_{FoV}(\cdot)$ stands for the distance from the nearest border of the FoV. This distance can be computed according to equations

$$\begin{aligned} d_{xy}(k) &= \sqrt{(x_c(k) - x_l(k))^2 + (y_c(k) - y_l(k))^2}, \\ \beta_{diff,h}(k) &= \min\{\alpha_{diff,h}(k), \pi - \alpha_{diff,h}(k)\}, \\ \beta_{diff,v}(k) &= \min\{\alpha_{diff,v}(k), \pi - \alpha_{diff,v}(k)\}, \\ d_{FoV,xy}(k) &= d_{xy}(k) \sin\left(\beta_{diff,h}(k) - \frac{AoV_h}{2}\right), \\ d_{FoV,z}(k) &= \mathrm{dist}(P_l(k), P_c(k)) \sin\left(\beta_{diff,v}(k) - \frac{AoV_v}{2}\right), \\ d_{FoV}(k) &= \sqrt{d_{FoV,z}(k)^2 + d_{FoV,xy}(k)^2} - r_d, \end{aligned} \tag{14}$$

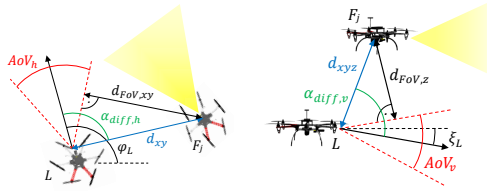
where $P_l(k) = [x_l, y_l, z_l]^T$ and $P_c(k) = [x_c, y_c, z_c]^T$ is the position of UAV carrying light and UAV carrying camera at the time corresponding to k-th transition point. AoV_h and AoV_v are horizontal and vertical angles of the camera FoV, $d_{FoV,xy}(k)$ is the distance to the nearest vertical border of FoV, $d_{FoV,z}(k)$ is the distance to the nearest horizontal border of FoV, and r_d marks the radius of the UAV. $\alpha_{diff,h}(k)$ and $\alpha_{diff,v}(k)$ stand for the angle between the nearest vertical respectively horizontal border of the FoV and connecting line between UAV carrying camera and UAV providing light. $\beta_{diff,h}(k)$ and $\beta_{diff,v}(k)$ are equivalent to $\alpha_{diff,h}(k)$ and $\alpha_{diff,v}(k)$, but besides the FoV of the camera, they include also the FoV of the virtual camera pointed in the exact opposite direction than the real camera. With this alteration, the J_{rti} penalizes not only the occlusion caused by the UAV carrying the light but also the shadows visible in the FoV caused by lighting from behind the UAV carrying the camera, which is important for the RTI image processing. The graphical illustration of symbols used in equation (14) is shown in Fig. 5.

The set of nonlinear constraints $g_p(\cdot) \leq 0$ can be broken down into the following constraints

$$\begin{aligned} g_c(\mathcal{U}_p(k)) &\leq 0, \forall k \in \{1, ..., N\}, \\ g_{obs}(P_l(k), \mathcal{O}(t)) &\leq 0, \forall k \in \{1, ..., N\}, \\ g_{rti}(P_l(k), \mathcal{O}(t), \psi_c(k)) &\leq 0, \forall k \in \{1, ..., N\}, \end{aligned} \tag{15}$$

where $\psi_c(k)$ stands for the configuration of the UAV carrying camera, $g_c(\cdot)$ includes the limitations on control inputs, $g_{obs}(\cdot)$ defines the infeasibility of trajectories colliding with obstacles, and $g_{rti}(\cdot)$ complements the objective function J_{rti}

This is the author's version of an article that has been published. Changes were made to this version by the publisher prior to publication.
The final version of paper is available at https://doi.org/10.1109/LRA.2020.2970646



(a) top view (b) side view

Fig. 5: Graphical illustration of meaning of particular symbols used in equations (14) for computation of part of the objective function penalizing the occlusion caused by the UAV carrying a light

by defining the entire FoV as an unfeasible region.

In a similar manner, the process of finding the optimal sequence of orientation control inputs $\mathcal{U}_o(t)$ on the horizon of length N can be defined as the quadratic constrained optimization task with the objective function $J_o(\cdot)$ and set of nonlinear constraints $g_o(\cdot)$ as

$$\begin{aligned} \mathcal{U}_o(t)^* &= \arg\min J_o(\mathcal{U}_o(t)), \\ &s.\ t.\ g_o(u_o(k), O_j(k)) \leq 0, \forall k \in \{1, ..., N\}. \end{aligned} \tag{16}$$

The objective function $J_o(\cdot)$ consists of two parts

$$J_o = \zeta J_{or} + \kappa J_{co}, \tag{17}$$

where J_{or} is the part penalizing the deviation from the desired orientation of light, J_{co} stands for the part penalizing the fast changes in consequent control inputs $u_o(\cdot)$, and ζ and κ are weights used for scaling of parts of the objective function $J_o(\cdot)$. The set of nonlinear constraints $g_o(\cdot) \leq 0$, $\forall k \in \{1, ..., N\}$ can be split into the following constraints

$$\begin{aligned} g_{co}(u_o(k)) \leq 0, \forall k \in \{1, ..., N\}, \\ g_{or}(O_l(k)) \leq 0, \forall k \in \{1, ..., N\}, \end{aligned} \tag{18}$$

where $O_l(\cdot)$ is the orientation of the light carried by the UAV, $g_{co}(\cdot)$ stands for the constraints introducing the limits on control inputs, and $g_{or}(\cdot)$ introduces the limitations on angles that define the orientation of the light.

V. EXPERIMENTAL RESULTS

A. Performance of generation of lighting positions sequence

The purpose of this section is to qualitatively and quantitatively compare algorithm SPPA, FIB-LKH and their combination which applies SPPA part for the generation of the desired lighting positions and LKH solver for finding a path connecting these positions (further referenced as LKH). The test was performed on the testing case of 10000 samples, each with randomly chosen parameters $\lambda_{v,min}, \lambda_{v,max}, \lambda_{h,min}, \lambda_{h,max}, d_l, v_s$, and initial position of the UAV carrying the light P_i.

The quality of solutions was compared regarding time requirements and the length of the resulting path. Concerning the CPU time, the SPPA is faster than the others. However, since the total CPU time needed by any method does not exceed 0.5 s for all considered problems (computed on the single-core CPU Intel CORE i7 8250), this aspect is not important for our application. More significant parameter is the length of the paths produced by particular methods. Considering this criterion as the comparison value, SPPA is better or equals to LKH solution in 9% of test samples and is not longer by more than 50% in 98% of test samples. Paths generated by FIB-LKH approach are mostly the shortest among all methods. However, they do not fully exploit the borders of the defined scanning area (see Fig. 7c). More detailed results of the quantitative comparison are shown in Fig. 6. Examples of generated sequences by particular methods used for qualitative comparison are shown in Fig. 7.

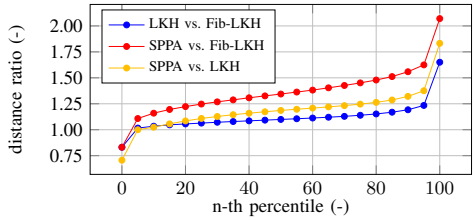
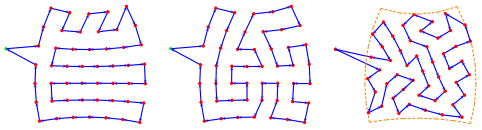


Fig. 6: Comparison of length of paths obtained by SPPA, LKH, and Fib-LKH approach



(a) (b) (c)

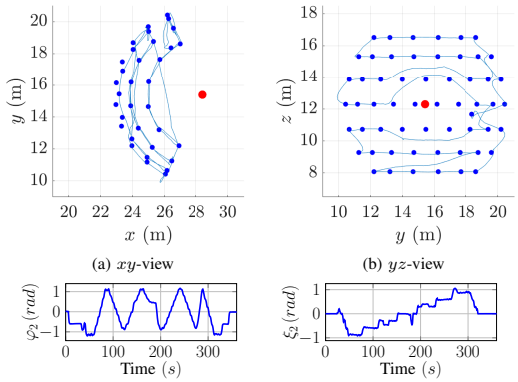
Fig. 7: Comparison of the solution obtained with SPPA (a), the solution generated by LKH method (b), and the solution generated by Fib-LKH approach (c). The orange dashed line marks the borders of the defined scanning area.

B. Verification of the overall RTI approach

The deployability of the SPPA method for the realization of RTI scanning, which was chosen together with experts from the Czech National Heritage Institute, who are potential end users of the proposed system, as the best variant for its real deployment, was verified through several experiments in the realistic robotic simulator Gazebo and within real-world experiments deploying two autonomous UAVs in the interior of the Church of St. Mary Magdalene in Chlumín.

The presented simulation in which the RTI scanning procedure is performed on the statue situated above the altar leads to the generation of 56 RTI positions and the resulting trajectory of the overall length 110.55 m. The set of generated points together with the trajectory flown by the UAV carrying the light are shown in Figure 8. In compliance with the theory presented in section IV, the UAV carrying light stops at each reachable RTI position and waits until an image is taken by the UAV carrying the camera. In this way, the system collects 56 images of the scanned object under various lighting conditions. The images are then registered to each other to compensate for the motion of UAV carrying a camera during the scanning process. Based on the registered images and the file containing the information about corresponding lighting directions, the PTM representation of the image is computed with the use of program PTM Fitter.

This is the author's version of an article that has been published. Changes were made to this version by the publisher prior to publication.
The final version of paper is available at https://doi.org/10.1109/LRA.2020.2970646



(a) xy-view (b) yz-view

(c) yaw - light (d) pitch - light

Fig. 8: The generated RTI positions and the trajectory flown by the UAV carrying the light during the RTI scanning procedure. The blue dots mark particular RTI positions, the red dot marks the position of the scanned object, and the blue line shows the trajectory.

The main advantage of obtaining the PTM from the set of images is that the image can be displayed under arbitrary lighting conditions. Since this result can be hardly presented within the printed work, the resulting PTM representation of the scanned object, obtained from the images taken by an onboard camera, is shown in the video available at http://mrs.felk.cvut.cz/papers/rti2020ral.

The real experiment was adapted to fit into the restricted space of the church in Chlumín. To enable the comparison of results of the proposed method and H-RTI, the object of interest (part of the pulpit) was chosen in the height accessible by people and it was illuminated from the same 12 positions (see Fig. 3b) by two different approaches - with the camera carried by an autonomous UAV (see Fig. 2) and with the camera mounted on a static tripod (see Fig. 12c). The latter approach eliminates the imprecision caused by the camera motion and hence enables the objective comparison of the results obtained from the same set of images with lighting vectors computed from the reflections on the black ball (H-RTI) and from the position of the light-carrying UAV provided by the application-tailored localization system [25]. The images generated based on the PTM representation of the scanned object are shown in Fig. 9 and in the video available at http://mrs.felk.cvut.cz/papers/rti2020ral.

(a) upper-left light (b) left light (c) bottom-right light

Fig. 9: Presentation of the PTM representation of the scanned object obtained from images taken by UAV during the RTI experiment performed in Church of St. Mary Magdalene in Chlumín. For video see http://mrs.felk.cvut.cz/papers/rti2020ral.

C. Dependence of PTM quality on precision of localization

To examine how the precision of localization affects the quality of the resulting PTM, a simulation-based quantitative comparison was conducted. The whole RTI procedure was performed on a lion statue with localization error sampled from the normal distribution with zero mean and multiple distinct values of standard deviation. The normal map obtained using SPPA (60 positions) and a modelled localization error is compared to the normal map obtained with SPPA (360 positions and zero localization error) used as ground truth. The results of this comparison are presented in Fig. 10. The average difference from the ground truth normals for the normal map obtained for the precision of localization presented in [25] is 0.026 rad (see Fig. 11 for details). This value is lower than the average difference caused by the misplacement of the reflective ball with respect to the center of the scanned object within the H-RTI procedure, which is unavoidable in this method.

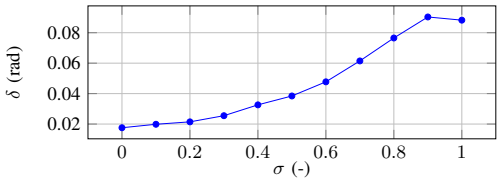
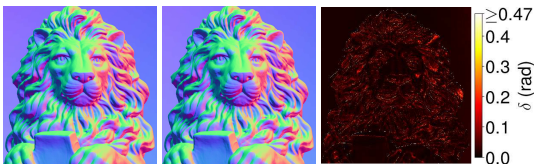


Fig. 10: The dependence of the average error in normals δ on the simulated localization error represented by $\mathcal{N}(0, \sigma^2)$ for positional error (m) and $\mathcal{N}(0, (\frac{2\pi\sigma}{36})^2)$ for orientation error (rad). Values of δ for a particular σ is computed as an average result of 20 experiments.



(a) (b) (c)

Fig. 11: Comparison of the normal map obtained with SPPA (360 positions and zero localization error) (a) used as a ground truth and normal map obtained with SPPA (60 positions and localization error (m) modeled by $\mathcal{N}(0,\ 0.09)$ for position and $\mathcal{N}(0,\ 0.003)$ for orientation) (b). Figure (c) shows the size of angle between normal vectors in maps (a) and (b) for particular pixels.

D. Comparison of the proposed approach with H-RTI

For comparison of the proposed approach and H-RTI, results of the RTI method in the form of normal maps are presented in Fig. 12. These two normal maps were obtained with lighting vectors computed by H-RTI method and with lighting vectors computed based on the pose of UAV obtained by the application-tailored localization system. Although the ground truth measurement is not available, we can, based on the known structure of the pulpit, claim that the results obtained with H-RTI method are more precise especially in the surroundings of the reflective ball.

However, the proposed method has an undeniable advantage in realization of the RTI method in hardly accessible places.

This is the author's version of an article that has been published. Changes were made to this version by the publisher prior to publication.
The final version of paper is available at https://doi.org/10.1109/LRA.2020.2970646

Moreover, under the condition of sufficiently precise localization, which is achieved by the applied localization system [25], the determination of lighting vectors is more precise than its detection from reflections on the ball, which cannot be placed directly in the center of a scanned object. The main drawbacks with respect to manually performed RTI lie in the inability to eliminate any camera motion. This issue is partially solved by the image registration process, however, on the high level of details, the imperfections of the alignment can cause unsharpness in images generated from the PTM representation. The camera motion, together with the high exposure time required in dark conditions, also causes the blur in images taken by the camera. However, this problem can be suppressed by the mechanical stabilization of the camera or by use of light source with higher power output, which enables the reduction of exposure time.

(a) (b) (c)

Fig. 12: Comparison of the normal map obtained with SPPA (12 positions) with lighting vectors computed by H-RTI method (a) and with lighting vectors computed based on the pose of UAV (b) obtained by the application-tailored localization system [25]. The setup for this experiment is shown in (c).

VI. CONCLUSION

The method for the realization of Reflectance Transformation Imaging with the use of a team of autonomous cooperative UAVs is described in this paper. The method is designed for two multi-rotor UAVs equipped with a camera and light source that are capable of self-localization within a given map of an environment. Three approaches to the generation of sequences of RTI positions are presented, but only one was approved by representatives of the heritage institute for its deployment in historical objects. This solution includes self-designed methods for generation of human-predictable trajectories to enable simple monitoring of correct behavior of particular UAVs by safety pilots, while preserving an effort to generate short trajectories. The compromise between these two criteria enables the safe deployment of the system in real-world scenarios. The main advantage of the proposed solution in comparison to already existing methods is the ability to perform the RTI scanning procedure in places that are hardly accessible or even inaccessible to humans.

The proposed approach was integrated into the system for documentation of historical buildings proposed in [10] and its practical applicability was tested in numerous experiments in interiors of churches in realistic simulator Gazebo and within the real experiment in Church of St. Mary Magdalene in Chlumín. Outputs of these tests were evaluated by experts from the field of historical science, who found the results comparable with the results produced by already existing methods, which are limited to accessible locations.

REFERENCES

[1] H. Mytum *et al.*, "The application of reflectance transformation imaging (rti) in historical archaeology," *Historical Archaeology*, vol. 52, no. 2, pp. 489–503, 2018.
[2] D. Selmo *et al.*, "Underwater reflectance transformation imaging: a technology for in situ underwater cultural heritage object-level recording," *Journal of Electronic Imaging*, vol. 26, no. 1, pp. 1–18, 2017.
[3] J. Miles *et al.*, "New applications of photogrammetry and reflectance transformation imaging to an easter island statue," *Antiquity*, vol. 88, no. 340, pp. 596–605, 2014.
[4] H. Mytum *et al.*, "Reflectance transformation imaging (rti) : Capturing gravestone detail via multiple digital images," in *Association for Gravestone Studies Quarterly*, 2017.
[5] J. Valcarcel Andrs *et al.*, "Applications of reflectance transformation imaging for documentation and surface analysis in conservation," *Internetional Journal of Conservation Science*, no. 4, pp. 535–548, 2013.
[6] D. Saunders *et al.*, "Reflectance transformation imaging and imagej: Comparing imaging methodologies for cultural heritage artefacts," in *EVA*, 2017.
[7] Y. H. Kim *et al.*, "Reflectance transformation imaging method for large-scale objects," in *CGiV*, 2016.
[8] A. Cosentino, "Macro photography for reflectance transformation imaging: A practical guide to the highlights method," *e-conservation Journal*, no. 1, pp. 70–85, 2013.
[9] R. Swinbank *et al.*, "Fibonacci grids: A novel approach to global modelling," *Quarterly Journal of the Royal Meteorological Society*, vol. 132, pp. 1769 – 1793, 2006.
[10] M. Saska *et al.*, "Documentation of dark areas of large historical buildings by a formation of unmanned aerial vehicles using model predictive control," in *IEEE ETFA*, 2017.
[11] M. Roberts *et al.*, "Submodular trajectory optimization for aerial 3d scanning," in *IEEE ICCV*, 2017.
[12] J. P. Fentanes *et al.*, "Algorithm for efficient 3d reconstruction of outdoor environments using mobile robots," in *IEEE IROS*, 2011.
[13] B. Adler *et al.*, "Finding next best views for autonomous uav mapping through gpu-accelerated particle simulation," in *IEEE/RSJ IROS*, 2013.
[14] P. S. Blaer *et al.*, "Data acquisition and view planning for 3-d modeling tasks," in *IEEE/RSJ IROS*, 2007.
[15] M. Nieuwenhuisen *et al.*, "Chimneyspector: Autonomous mav-based indoor chimney inspection employing 3d laser localization and textured surface reconstruction," in *IEEE ICUAS*, 2017.
[16] N. Hallermann *et al.*, "Vision-based monitoring of heritage monuments unmanned aerial systems (uas) for detailed inspection and high-accurate survey of structures," in *STREMAH*, 2015.
[17] H. Qin *et al.*, "Autonomous exploration and mapping system using heterogeneous uavs and ugvs in gps-denied environments," *IEEE Transactions on Vehicular Technology*, vol. 68, no. 2, pp. 1339–1350, 2019.
[18] Y. Lyu *et al.*, "Simultaneously multi-uav mapping and control with visual servoing," in *IEEE ICUAS*, 2015.
[19] M. Saska *et al.*, "Coordination and Navigation of Heterogeneous MAV & UGV Formations Localized by a "hawk-eye"-like Approach Under a Model Predictive Control Scheme," *International Journal of Robotics Research*, vol. 33, no. 10, pp. 1393–1412, 2014.
[20] V. Spurný *et al.*, "Complex manoeuvres of heterogeneous mav-ugv formations using a model predictive control," in *MMAR*, 2016.
[21] M. Saska *et al.*, "Predictive control and stabilization of nonholonomic formations with integrated spline-path planning," *Robotics and Autonomous Systems*, 2015.
[22] T. Báča *et al.*, "Autonomous landing on a moving vehicle with an unmanned aerial vehicle," *Journal of Field Robotics*, vol. 36, pp. 874–891, 2019.
[23] T. Malzbender *et al.*, "Polynomial Texture Maps," in *CGIT*, 2001.
[24] T. Báča *et al.*, "Model predictive trajectory tracking and collision avoidance for reliable outdoor deployment of unmanned aerial vehicles," in *IEEE/RSJ IROS*, 2018.
[25] P. Petráček *et al.*, "Dronument: Reliable deployment of unmanned aerial vehicles in dark areas of large historical monuments," *IEEE RA-L*, In Press: Accepted for publication on January 9, 2020.
[26] E. B. Saff *et al.*, "Distributing many points on a sphere," *The Mathematical Intelligencer*, vol. 19, no. 1, pp. 5–11, 1997.
[27] S. Lin *et al.*, "An effective heuristic algorithm for the traveling-salesman problem," *Operations Research*, vol. 21, no. 2, pp. 498–516, 1973.
[28] M. Dellepiane *et al.*, "High quality ptm acquisition: Reflection transformation imaging for large objects," in *International Symposium on VAST*, 2006.

Chapter 9
Conclusion

In this thesis, several aspects required for the deployment of cooperating autonomous UAVs were tackled. The emphasis of this work is on designing a new methodology for various real-world robotic scenarios. At the time of working on this thesis, the standard in the multi-robot UAV system community was to use simulated testing or short isolated experiments under laboratory conditions. These experiments would rely on motion capture systems with no external disturbances, using stable communication infrastructure without interference. Compared to these approaches, real-world deployment raised additional constraints and fundamental research challenges that were difficult to solve. In order to overcome such challenges, robotic systems tailored for these purposes should mainly rely only on data obtained from sensors onboard the robot, which made it necessary for a novel theory to be designed. A robotic system for real-world applications requires a reliable high-level planning architecture that is able to recover from robot failures or from a malfunction of its system parts. This is essential for UAV and especially for multi-UAV systems. For this, we have proposed a novel Failure recovery and Synchronization jobs Manager (FSM). Communication between robots is required to fully access the capabilities of the deployed multi-robot team. In addition to the failures of sensors and actuators, the technical requirements for working communication infrastructure are currently still a bottleneck. Therefore, we made the FSM considering different strategies based on the availability of communication infrastructure. This allows us to maximize the contribution of multi-robot system and their tight cooperation whenever it is enabled.

This thesis was mainly inspired by three real-world scenarios: autonomous delivery of objects by a team of UAVs, autonomous aerial surveys of historical building interiors, and autonomous firefighting. The delivery and firefighting scenarios were motivated by challenges in the Mohamed Bin Zayed International Robotics Challenge (MBZIRC) 2017 and 2020 competitions. The MBZIRC competition is demanding not only due to the selected tasks being on the edge of the state of the art in robotics, but also because it allows only the short preparation time of 15 min before trials. The short time for system preparation goes hand in hand with requirements of most of the robotic scenarios, especially for search & rescue missions where every second counts. The scientific foundations presented in this thesis achieved first place in the delivery challenge at MBZIRC 2017 and first place in the Grand Challenge of MBZIRC 2020, which was awarded as the total winner of the competition among the best teams worldwide. These achievements highlight the proposed methodology's reliability and potential for fast deployment in real-world conditions.

The second part of the thesis was focused on motion planning for multiple cooperating UAVs. We proposed a novel sampling-based motion planning method designed for solving a cooperative transportation task by a UAV pair in environments with obstacles. The proposed concept aims to solve the given task by planning a path that prefers to maintain the shape of the pair in a configuration that is close to the one that is desired for carrying, as long as it is enabled by environmental constraints while allowing for temporary deviation when necessary. This methodology was again targeted for real-world deployment with onboard sensors only. Furthermore, we showed that a team of cooperating UAVs is suited for visual documentation

and inspection of the interiors of buildings. For that, we proposed a self-stabilized formation of UAVs employed for filming and visual inspection in dark conditions, where one of the UAVs carries a camera and the neighboring UAVs carries a light source. The proposed system enables setting the direction of the light sources dependently on the position of the camera, which makes it possible to fully autonomously realize lighting methods that are often used by historians and restorers to manually inspect the interiors of historical monuments.

9.1 Future work

In my opinion, the most interesting scenario, from the considered ones in the thesis, is autonomous aerial surveys of historical building interiors. In future work, our goal will be to exploit different techniques used by historians and restorers that can be applied using cooperating UAVs. For example, promising sensors seem to be multispectral cameras that are being nowadays smaller, lighter, and more available. These cameras combined with lights that work in different parts of the spectrum than visible ones could for example help recover covered painting under layers of different paint.

Chapter A
Author's publications

Listed below are all publications of the author. Each citation is displayed with a percentage contribution of the author and the number of citations based on Web of Science (WoS), Scopus, and Google Scholar (GS). Journal articles also contain information about the Impact Factor (IF) by Thomson Reuters, if it exists. The core publications of this thesis are referenced with [*c] and the other author's publications with [*a]. The data were gathered on August 18, 2021.

A.1 Thesis-related author's publications

A.1.1 Thesis core publications

Thesis core articles in peer-reviewed journals with Impact Factor

[1c] **V. Spurny**, V. Pritzl, V. Walter, M. Petrlik, T. Baca, P. Stepan, D. Zaitlik, and M. Saska, "Autonomous Firefighting Inside Buildings by an Unmanned Aerial Vehicle," *IEEE Access*, vol. 9, pp. 15 872–15 890, 2021
44% contribution, IF 3.745 (Q1 in Computer Science, Information Systems), citations: 1 in Scopus, 3 in GS.

[2c] G. Loianno, **V. Spurny**, J. Thomas, T. Baca, D. Thakur, D. Hert, R. Penicka, T. Krajnik, A. Zhou, A. Cho, M. Saska, and V. Kumar, "Localization, Grasping, and Transportation of Magnetic Objects by a team of MAVs in Challenging Desert-like Environments," *IEEE Robotics and Automation Letters*, vol. 3, no. 3, pp. 1576–1583, 2018
11% contribution, IF 3.608 (Q1 in Robotics), citations: 36 in WoS, 40 in Scopus, 74 in GS.

[3c] **V. Spurny**, T. Baca, M. Saska, R. Penicka, T. Krajnik, J. Thomas, D. Thakur, G. Loianno, and V. Kumar, "Cooperative Autonomous Search, Grasping, and Delivering in a Treasure Hunt Scenario by a Team of UAVs," *Journal of Field Robotics*, vol. 36, no. 1, 125–148, 2019
50% contribution, IF 4.345 (Q1 in Robotics), citations: 36 in WoS, 36 in Scopus, 81 in GS.

[4c] V. Kratky, P. Petracek, **V. Spurny**, and M. Saska, "Autonomous Reflectance Transformation Imaging by a Team of Unmanned Aerial Vehicles," *IEEE Robotics and Automation Letters*, vol. 5, no. 2, pp. 2302–2309, 2020
10% contribution, IF 3.608 (Q1 in Robotics), citations: 4 in WoS, 8 in Scopus, 14 in GS.

Thesis core conference proceedings in WoS

[5c] **V. Spurny**, M. Petrlik, V. Vonasek, and M. Saska, "Cooperative Transport of Large Objects by a Pair of Unmanned Aerial Systems using Sampling-based Motion Planning," in *IEEE International Conference on Emerging Technologies and Factory Automation (ETFA)*, 2019, pp. 955–962
52% contribution, citations: 1 in WoS, 2 in Scopus, 17 in GS.

[6c] M. Saska, V. Kratky, **V. Spurny**, and T. Baca, "Documentation of Dark Areas of Large Historical Buildings by a Formation of Unmanned Aerial Vehicles using Model Predictive Control," in *IEEE International Conference on Emerging Technologies and Factory Automation (ETFA)*, 2017, pp. 1–8
25% contribution, citations: 14 in WoS, 12 in Scopus, 51 in GS, (BEST PAPER AWARD).

A.1.2 Other thesis-related publications

Other thesis-related articles in peer-reviewed journals with Impact Factor

[7a] M. Saska, T. Krajnik, V. Vonasek, Z. Kasl, **V. Spurny**, and L. Preucil, "Fault-Tolerant Formation Driving Mechanism Designed for Heterogeneous MAVs-UGVs Groups," *Journal of Intelligent and Robotic Systems*, vol. 73, no. 1-4, pp. 603–622, 2014
IF 2.259 (Q3 in Robotics), citations: 23 in WoS, 24 in Scopus, 85 in GS.

[8a] M. Saska, **V. Spurny**, and V. Vonasek, "Predictive control and stabilization of nonholonomic formations with integrated spline-path planning," *Robotics and Autonomous Systems*, vol. 75, pp. 379–397, 2016
IF 2.825 (Q2 in Robotics), citations: 19 in WoS, 21 in Scopus, 58 in GS.

[9a] T. Baca, P. Stepan, **V. Spurny**, D. Hert, R. Penicka, M. Saska, J. Thomas, G. Loianno, and V. Kumar, "Autonomous Landing on a Moving Vehicle with an Unmanned Aerial Vehicle," *Journal of Field Robotics*, vol. 36, pp. 874–891, 5 2019
IF 4.345 (Q1 in Robotics), citations: 20 in WoS, 23 in Scopus, 62 in GS.

[10a] T. Baca, M. Petrlik, M. Vrba, **V. Spurny**, R. Penicka, D. Hert, and M. Saska, "The MRS UAV System: Pushing the Frontiers of Reproducible Research, Real-world Deployment, and Education with Autonomous Unmanned Aerial Vehicles," *Journal of Intelligent & Robotic Systems*, vol. 102, no. 26, pp. 1–28, 1 2021
IF 2.259 (Q3 in Robotics), citations: 6 in WoS, 6 in Scopus, 17 in GS.

[11a] D. Horla, W. Giernacki, T. Baca, **V. Spurny**, and M. Saska, "AL-TUNE: In-flight tuning of an altitude controller for UAVs," *Journal of Intelligent & Robotic Systems*, vol. 103, no. 5, pp. 1–16, 1 2021
IF 2.259 (Q3 in Robotics).

Other thesis-related articles in peer-reviewed journals with Impact Factor under review process

[12a] M. Vrba, Y. Stasinchuk, T. Baca, **V. Spurny**, M. Petrlik, D. Hert, D. Zaitlik, and M. Saska, "Autonomous Capturing of Agile Flying Objects using UAVs: The MBZIRC 2020 Challenge," Aug. 2021, Submitted to *Robotics and Autonomous Systems*
IF 3.12 (Q2 in Robotics).

[13a] T. Baca, R. Penicka, P. Stepan, M. Petrlik, **V. Spurny**, D. Hert, and M. Saska, "Autonomous Cooperative Wall Building by a Team of Unmanned Aerial Vehicles in the MBZIRC 2020 Competition," Dec. 2020, Submitted to *Robotics and Autonomous Systems*
IF 3.12 (Q2 in Robotics).

Other thesis-related articles in recently established peer-reviewed journals (no Impact Factor)

[14a] V. Walter, **V. Spurny**, M. Petrlik, T. Baca, D. Zaitlik, L. Demkiv, and M. Saska, "Extinguishing of Real Fires by Fully Autonomous MAVs in the MBZIRC 2020 Competition," May 2021, Accepted to *Field Robotics*.

[15a] Y. Stasinchuk, M. Vrba, T. Baca, **V. Spurny**, M. Petrlik, D. Hert, D.Zaitlik, and M.Saska, "A Multi-MAV System For Autonomous Multiple Targets Elimination in the MBZIRC 2020 competition," May 2021, Accepted to *Field Robotics*.

[16a] K. Jindal, A. Wang, D. Thakur, A. Zhou, **V.Spurny**, V.Walter, G. Broughton, T. Krajnik, M. Saska, and G. Loianno, "Design and Deployment of an Autonomous Unmanned Ground Vehicle for Urban Firefighting Scenarios," Apr. 2021, Accepted to *Field Robotics*.

[17a] T. Roucek, M. Pecka, P. Cizek, T. Petricek, J. Bayer, V. Salansky, D. Hert, M. Petrlik, T. Baca, **V. Spurny**, V. Kratky, P. Petracek, D. Baril, M. Vaidis, V. Kubelka, F. Pomerleau, J. Faigl, K. Zimmermann, M. Saska, T. Svoboda, and T. Krajnik, "System for multi-robotic exploration of underground environments CTU-CRAS-NORLAB in the DARPA Subterranean Challenge," Jun. 2021, Accepted to *Field Robotics*.

Other thesis-related conference proceedings in WoS

[18a] V. Walter, **V. Spurny**, M. Petrlik, T. Baca, D. Zaitlik, and M. Saska, "Extinguishing of Ground Fires by Fully Autonomous UAVs Motivated by the MBZIRC 2020 Competition," in *International Conference on Unmanned Aircraft Systems (ICUAS)*, 2021, pp. 787–793.

[19a] M. Saska, **V. Spurny**, and L. Preucil, "Trajectory Planning and Stabilization for Formations Acting in Dynamic Environments," in *Progress in Artificial Intelligence*, Springer, 2013, pp. 319–330
citations: 2 in WoS, 3 in Scopus, 8 in GS.

[20a] **V. Spurny**, T. Baca, and M. Saska, "Complex manoeuvres of heterogeneous MAV-UGV formations using a model predictive control," in *International Conference on Methods and Models in Automation and Robotics (MMAR)*, 2016, pp. 998–1003
citations: 7 in WoS, 11 in Scopus, 52 in GS.

[21a] M. Vrba, J. Pogran, V. Pritzl, **V. Spurny**, and M. Saska, "Real-time localization of transmission sources using a formation of micro aerial vehicles," in *IEEE International Conference on Real-time Computing and Robotics (RCAR)*, 2019, pp. 203–208
citations: 1 in Scopus, 1 in GS.

[22a] T. Roucek, M. Pecka, P. Cizek, T. Petracek, J. Bayer, V. Salansky, D. Hert, M. Petrlik, T. Baca, **V. Spurny**, F. Pomerleau, V. Kubelka, J. Faigl, K. Zimmermann, M. Saska, T. Svoboda, and J. Krajnik, "DARPA Subterranean Challenge: Multi-robotic exploration of underground environments," in *Modelling and Simulation for Autonomous Systems (MESAS)*, vol. 11995, Springer, 2019, pp. 274–290
citations: 7 in WoS, 6 in Scopus, 21 in GS.

[23a] J. Faigl, P. Vana, M. Saska, T. Baca, and **V. Spurny**, "On solution of the Dubins touring problem," in *IEEE European Conference on Mobile Robots (ECMR)*, 2017, pp. 1–6
citations: 11 in WoS, 9 in Scopus, 23 in GS.

[24a] Y. Stasinchuk, M. Vrba, M. Petrlik, T. Baca, **V. Spurny**, D. Hert, D. Zaitlik, T. Nascimento, and M. Saska, "A Multi-UAV System for Detection and Elimination of Multiple Targets," Accepted to *IEEE International Conference on Robotics and Automation (ICRA)*, Jun. 2021.

Chapter B

SCI Citations of author's work

Citations of the author's work were extracted from the Web of Science. First-order and second-order self-citations are excluded. The data were gathered on August 18, 2021.

[7a] M. Saska, T. Krajnik, V. Vonasek, Z. Kasl, **V. Spurny**, and L. Preucil, "Fault-Tolerant Formation Driving Mechanism Designed for Heterogeneous MAVs-UGVs Groups," *Journal of Intelligent and Robotic Systems*, vol. 73, no. 1-4, pp. 603–622, 2014

- M. A. Kamel, X. Yu, and Y. Zhang, "Real-Time Fault-Tolerant Formation Control of Multiple WMRs Based on Hybrid GA-PSO Algorithm," *IEEE Transactions on Automation Science and Engineering*, vol. 18, no. 3, 1263–1276, Jan. 2021.
- C. Liu, B. Jiang, R. J. Patton, and K. Zhang, "Decentralized Output Sliding-mode Fault-tolerant Control For Heterogeneous Multiagent Systems," *IEEE Transactions on Cybernetics*, vol. 50, no. 12, 4934–4945, Jan. 2020.
- C. Liu, B. Jiang, and K. Zhang, "Adaptive Fault-tolerant H-infinity Output Feedback Control for Lead-wing Close Formation Flight," *IEEE Transactions on Systems Man Cybernetics-systems*, vol. 50, no. 8, 2804–2814, Jan. 2020.
- G. C. Karras and G. K. Fourlas, "Model Predictive Fault Tolerant Control for Omni-directional Mobile Robots," *Journal of Intelligent & Robotic Systems*, vol. 97, no. 3-4, 635–655, Jan. 2020.
- Z. Liu, Y. Zhang, C. Yuan, L. Ciarletta, and D. Theilliol, "Collision Avoidance and Path Following Control of Unmanned Aerial Vehicle In Hazardous Environment," *Journal of Intelligent & Robotic Systems*, vol. 95, no. 1, 193–210, Jan. 2019.
- A. D. Dang, H. M. La, T. Nguyen, and J. Horn, "Formation Control for Autonomous Robots With Collision and Obstacle Avoidance Using A Rotational and Repulsive Force-based Approach," *International Journal of Advanced Robotic Systems*, vol. 16, no. 3, Jan. 2019.
- X. Ma, W. Dong, and B. Li, "Comprehensive Fault-tolerant Control of Leader-follower Unmanned Aerial Vehicles (UAVs) Formation," *International Journal of Robotics & Automation*, vol. 34, no. 6, 695–712, 2019.
- C. Liu, B. Jiang, and K. Zhang, "Output Feedback Fault-tolerant Control for Lead-wing Close Formation Flight," in *12th Asian Control Conference (ASCC)*, 2019, 295–300.
- S. Suzuki, "Recent Researches on Innovative Drone Technologies In Robotics Field," *Advanced Robotics*, vol. 32, no. 19, Si, 1008–1022, Jan. 2018.
- Z. Liu, C. Yuan, and Y. Zhang, "Active Fault-tolerant Control of Unmanned Quadrotor Helicopter Using Linear Parameter Varying Technique," *Journal of Intelligent & Robotic Systems*, vol. 88, no. 2-4, 1, Si, 415–436, Jan. 2017.
- Z. Liu, C. Yuan, Y. Zhang, and J. Luo, "A Learning-based Fault Tolerant Tracking Control of An Unmanned Quadrotor Helicopter," *Journal of Intelligent & Robotic Systems*, vol. 84, no. 1-4, Si, 145–162, Jan. 2016.
- V. Cichella, R. Choe, S. B. Mehdi, E. Xargay, N. Hovakimyan, V. Dobrokhodov, I. Kaminer, A. M. Pascoal, and A. P. Aguiar, "Safe Coordinated ManEUVering of Teams of Multirotor Unmanned Aerial Vehicles A Cooperative Control Framework for Multivehicle, Time-critical Missions," *IEEE Control Systems Magazine*, vol. 36, no. 4, 59–82, Jan. 2016.
- V. Cichella, I. Kaminer, V. Dobrokhodov, E. Xargay, R. Choe, N. Hovakimyan, A. P. Aguiar, and A. M. Pascoal, "Cooperative Path Following of Multiple Multirotors Over Time-varying Networks," *IEEE Transactions on Automation Science and Engineering*, vol. 12, no. 3, 945–957, Jan. 2015.

[8a] M. Saska, **V. Spurny**, and V. Vonasek, "Predictive control and stabilization of nonholonomic formations with integrated spline-path planning," *Robotics and Autonomous Systems*, vol. 75, pp. 379–397, 2016

- A. S. Lafmejani and S. Berman, "Nonlinear MPC for collision-free and deadlock-free navigation of multiple nonholonomic mobile robots," *Robotics and Autonomous Systems*, vol. 141, Jan. 2021.
- X. Yang, W. Wang, and P. Huang, "Distributed Optimal Consensus With Obstacle Avoidance Algorithm Of Mixed-order UAVs-usvs-uuvs Systems," *Isa Transactions*, vol. 107, 270–286, Jan. 2020.
- I. Akli, "Experimental Analysis of the Performance of Mobile Manipulators Executing Generalized Point-to-point Tasks," *Arabian Journal for Science and Engineering*, vol. 45, no. 3, 2327–2339, Jan. 2020.
- Y. Zhang, B. Gao, L. Guo, H. Guo, and M. Cui, "A Novel Trajectory Planning Method for Automated Vehicles Under Parameter Decision Framework," *IEEE Access*, vol. 7, 88264–88274, 2019.
- X. Sun, G. Wang, Y. Fan, D. Mu, and B. Qiu, "A Formation Collision Avoidance System for Unmanned Surface Vehicles With Leader-follower Structure," *IEEE Access*, vol. 7, 24691–24702, 2019.
- M. Murillo, G. Sanchez, L. Genzelis, and L. Giovanini, "A Real-time Path-planning Algorithm Based on Receding Horizon Techniques," *Journal of Intelligent & Robotic Systems*, vol. 91, no. 3-4, 445–457, Jan. 2018.
- F. Rahimi and R. M. Esfanjani, "Distributed Predictive Control for Formation of Networked Mobile Robots," in *6th Rsi International Conference on Robotics and Mechatronics (ICROM 2018)*, 2018, 70–75.
- R. Kala, "On Repelling Robotic Trajectories: Coordination In Navigation Of Multiple Mobile Robots," *Intelligent Service Robotics*, vol. 11, no. 1, 79–95, Jan. 2018.
- M. H. Yamchi and R. M. Esfanjani, "Distributed Predictive Formation Control of Networked Mobile Robots Subject To Communication Delay," *Robotics and Autonomous Systems*, vol. 91, 194–207, Jan. 2017.
- A. Rodriguez-ramos, C. Sampedro, H. Bavle, Z. Milosevic, A. Garcia-vaquero, and P. Campoy, "Towards Fully Autonomous Landing on Moving Platforms for Rotary Unmanned Aerial Vehicles," in *International Conference on Unmanned Aircraft Systems (ICUAS)*, 2017, 170–178.
- P. K. Das, H. S. Behera, S. Das, H. K. Tripathy, B. K. Panigrahi, and S. K. Pradhan, "A Hybrid Improved Pso-dv Algorithm for Multi-robot Path Planning In A Clutter Environment," *Neurocomputing*, vol. 207, 735–753, Jan. 2016.
- M. Kulich, J. Chudoba, and L. Preucil, "Practical Applications and Experiments With the Syrotek Platform," in *Modelling and Simulation for Autonomous Systems (MESAS)*, vol. 9055, 2015, 90–101.

[20a] **V. Spurny**, T. Baca, and M. Saska, "Complex manoeuvres of heterogeneous MAV-UGV formations using a model predictive control," in *International Conference on Methods and Models in Automation and Robotics (MMAR)*, 2016, pp. 998–1003

- A. Madridano, A. Al-Kaff, D. Martin, and A. de la Escalera, "Trajectory planning for multi-robot systems: Methods and applications," *Expert Systems With Applications*, vol. 173, Jan. 2021.
- W. T. Botelho, M. D. G. Bruno Marietto, E. D. L. Mendes, D. R. De Sousa, E. P. Pimentel, V. L. Da Silva, and T. Dos Santos, "Toward An Interdisciplinary Integration Between Multi-agents Systems And Multi-robots Systems: A CASE Study," *Knowledge Engineering Review*, vol. 35, 2020.

[6c] M. Saska, V. Kratky, **V. Spurny**, and T. Baca, "Documentation of Dark Areas of Large Historical Buildings by a Formation of Unmanned Aerial Vehicles using Model Predictive Control," in *IEEE International Conference on Emerging Technologies and Factory Automation (ETFA)*, 2017, pp. 1–8

- A. Wasik, P. U. Lima, and A. Martinoli, "A Robust Localization System for Multi-robot Formations Based on An Extension of A Gaussian Mixture Probability Hypothesis Density Filter," *Autonomous Robots*, vol. 44, no. 3-4, Si, 395–414, Jan. 2020.
- W. Giernacki, P. Kozierski, J. Michalski, M. Retinger, R. Madonski, and P. Campoy, "Bebop 2 Quadrotor As A Platform for Research and Education In Robotics and Control Engineering," in *International Conference on Unmanned Aircraft Systems (ICUAS)*, 2020, 1733–1741.
- A. V. Savkin and H. Huang, "Asymptotically Optimal Deployment of Drones for Surveillance And Monitoring," *Sensors*, vol. 19, no. 9, Jan. 2019.
- B. Sabetghadam, A. Alcantara, J. Capitan, R. Cunha, A. Ollero, and A. Pascoal, "Optimal Trajectory Planning for Autonomous Drone Cinematography," in *European Conference on Mobile Robots (ECMR)*, 2019.

[23a] J. Faigl, P. Vana, M. Saska, T. Baca, and **V. Spurny**, "On solution of the Dubins touring problem," in *IEEE European Conference on Mobile Robots (ECMR)*, 2017, pp. 1–6

- W. Pi and J. Zhou, "Multi-UAV Enabled Data Collection with Efficient Joint Adaptive Interference Management and Trajectory Design," *Electronics*, vol. 10, no. 5, Jan. 2021.
- A. Wolek, J. Mcmahon, B. R. Dzikowicz, and B. H. Houston, "The Orbiting Dubins Traveling Salesman Problem: Planning Inspection Tours for A Minehunting Auv," *Autonomous Robots*, vol. 45, no. 1, 31–49, Jan. 2021.

[2c] G. Loianno, **V. Spurny**, J. Thomas, T. Baca, D. Thakur, D. Hert, R. Penicka, T. Krajnik, A. Zhou, A. Cho, *et al.*, "Localization, Grasping, and Transportation of Magnetic Objects by a team of MAVs in Challenging Desert-like Environments," *IEEE Robotics and Automation Letters*, vol. 3, no. 3, pp. 1576–1583, 2018

- M. Basiri, J. Goncalves, J. Rosa, A. Vale, and P. Lima, "An Autonomous Mobile Manipulator To Build Outdoor Structures Consisting of Heterogeneous Brick Patterns," *Sn Applied Sciences*, vol. 3, no. 5, Jan. 2021.
- M. P. Arun, M. Satheesh, and J. E. R. Dhas, "Designing and Modelling of Delta Wing Genetic-based Prediction Model," *International Journal of Ambient Computing and Intelligence*, vol. 12, no. 1, 159–183, Jan. 2021.
- F. Real, A. R. Castano, A. Torres-gonzalez, J. Capitan, P. J. Sanchez-cuevas, M. J. Fernandez, M. Villar, and A. Ollero, "Experimental Evaluation of A Team of Multiple Unmanned Aerial Vehicles for Cooperative Construction," *IEEE Access*, vol. 9, 6817–6835, 2021.
- R. C. Shit, "Precise Localization for Achieving Next-generation Autonomous Navigation: State-of-the-art, Taxonomy and Future Prospects," *Computer Communications*, vol. 160, 351–374, Jan. 2020.
- S. Liu, W. Dong, Z. Ma, and X. Sheng, "Adaptive Aerial Grasping and Perching With Dual Elasticity Combined Suction Cup," *IEEE Robotics and Automation Letters*, vol. 5, no. 3, 4766–4773, Jan. 2020.
- G. Michieletto, A. Cenedese, L. Zaccarian, and A. Franchi, "Hierarchical Nonlinear Control for Multi-rotor Asymptotic Stabilization Based on Zero-moment Direction," *Automatica*, vol. 117, Jan. 2020.
- S. Patel, A. Sarabakha, D. Kircali, and E. Kayacan, "An Intelligent Hybrid Artificial Neural Network-based Approach For Control of Aerial Robots," *Journal of Intelligent & Robotic Systems*, vol. 97, no. 2, 387–398, Jan. 2020.

- K. Feng, W. Li, S. Ge, and F. Pan, "Packages Delivery Based on Marker Detection for UAVs," in *32nd Chinese Control and Decision Conference (CCDC)*, 2020, 2094–2099.
- H. C. Henninger, K. D. Von Ellenrieder, and S. C. Licht, "Energy-minimal Target Retrieval for Quadrotor UAVs: Trajectory Generation and Tracking," in *28th Mediterranean Conference on Control and Automation (MED)*, 2020, 727–732.
- I. H. B. Pizetta, A. S. Brandao, and M. Sarcinelli-filho, "Load Transportation By Quadrotors In Crowded Workspaces," *IEEE Access*, vol. 8, 223941–223951, 2020.
- M. Lopez Garcia and J. Martinez Carranza, "A First Cnn-based Approach Towards Autonomous Flight for Object Lifting," *Computacion Y Sistemas*, vol. 24, no. 3, 1219–1228, 2020.
- D. Sartori and W. Yu, "Experimental Characterization of A Propulsion System for Multi-rotor UAVs," *Journal of Intelligent & Robotic Systems*, vol. 96, no. 3-4, 529–540, Jan. 2019.
- D. K. D. Villa, A. S. Brandao, and M. Sarcinelli-filho, "A Survey on Load Transportation Using Multirotor UAVs," *Journal of Intelligent & Robotic Systems*, vol. 98, no. 2, 267–296, Jan. 2020.
- I. H. Beloti Pizetta, A. S. Brandao, and M. Sarcinelli-filho, "Avoiding Obstacles In Cooperative Load Transportation," *Isa Transactions*, vol. 91, 253–261, Jan. 2019.
- S. Hamaza, I. Georgilas, M. Fernandez, P. Sanchez, T. Richardson, G. Heredia, and A. Ollero, "Sensor Installation and Retrieval Operations Using An Unmanned Aerial Manipulator," *IEEE Robotics and Automation Letters*, vol. 4, no. 3, 2793–2800, Jan. 2019.
- I. H. Beloti Pizetta, A. Santos Brandao, and M. Sarcinelli-filho, "Cooperative Load Transportation Using Three Quadrotors," in *International Conference on Unmanned Aircraft Systems (ICUAS)*, 2019, 644–650.
- A. Sarabakha and E. Kayacan, "Online Deep Learning for Improved Trajectory Tracking of Unmanned Aerial Vehicles Using Expert Knowledge," in *International Conference on Robotics and Automation (ICRA)*, 2019, 7727–7733.
- H. Zhou, J. Lynch, and D. Zekkos, "Vision-based Precision Localization of UAVs for Sensor Payload Placement and Pickup for Field Monitoring Applications," in *Sensors and Smart Structures Technologies for Civil, Mechanical, And Aerospace Systems*, vol. 10970, 2019.
- R. Baehnemann, M. Pantie, M. Popovic, D. Schindler, M. Tranzatto, M. Kamel, M. Grimm, J. Widauer, R. Siegwart, and J. Nieto, "The ETH-MAV Team In the MBZInternational Robotics Challenge," *Journal of Field Robotics*, vol. 36, no. 1, Si, 78–103, Jan. 2019.
- M. Beul, M. Nieuwenhuisen, J. Quenzel, R. A. Rosu, J. Horn, D. Pavlichenko, S. Houben, and S. Behnke, "Team Nimbro At MBZIRC 2017: Fast Landing on A Moving Target and Treasure Hunting With A Team of Micro Aerial Vehicles," *Journal of Field Robotics*, vol. 36, no. 1, Si, 204–229, Jan. 2019.
- Y. Wu, J. Song, J. Sun, F. Zhu, and H. Chen, "Aerial Grasping Based on Vr Perception and Haptic Control," in *International Conference on Real-time Computing and Robotics (RCAR)*, 2018, 556–562.
- D. K. D. Villa, A. S. Brandao, and M. Sarcinelli-filho, "Load Transportation Using Quadrotors: A Survey of Experimental Results," in *International Conference on Unmanned Aircraft Systems (ICUAS)*, 2018, 84–93.

[9a] T. Baca, P. Stepan, **V. Spurny**, D. Hert, R. Penicka, M. Saska, J. Thomas, G. Loianno, and V. Kumar, "Autonomous Landing on a Moving Vehicle with an Unmanned Aerial Vehicle," *Journal of Field Robotics*, vol. 36, pp. 874–891, 5 2019

- D. Horla, M. Hamandi, W. Giernacki, and A. Franchi, "Optimal Tuning of the Lateral-Dynamics Parameters for Aerial Vehicles With Bounded Lateral Force," *IEEE Robotics and Automation Letters*, vol. 6, no. 2, 3949–3955, Jan. 2021.
- Y. Ding, B. Xin, and J. Chen, "A Review of Recent Advances In Coordination Between Unmanned Aerial And Ground Vehicles," *Unmanned Systems*, vol. 9, no. 2, 97–117, Jan. 2021.
- D. Horla, W. Giernacki, J. Cieslak, and P. Campoy, "Altitude Measurement-based Optimization of the Landing Process of UAVs," *Sensors*, vol. 21, no. 4, Jan. 2021.

- J. Xie, X. Peng, H. Wang, W. Niu, and X. Zheng, "UAV Autonomous Tracking and Landing Based on Deep Reinforcement Learning Strategy," *Sensors*, vol. 20, no. 19, Jan. 2020.
- D. Horla and J. Cieslak, "On Obtaining Energy-optimal Trajectories for Landing of UAVs," *Energies*, vol. 13, no. 8, Jan. 2020.
- Z. Boucek and M. Flidr, "Interpolating Control Based Trajectory Tracking," in *16th IEEE International Conference on Control, Automation, Robotics And Vision (ICARCV)*, 2020, 701–706.
- Z.-c. Xu, B.-b. Hu, B. Liu, X. D. Wang, and H.-t. Zhang, "Vision-based Autonomous Landing of Unmanned Aerial Vehicle on A Motional Unmanned Surface Vessel," in *39th Chinese Control Conference*, 2020, 6845–6850.
- A. Mohammadi, Y. Feng, C. Zhang, S. Rawashdeh, and S. Baek, "Vision-based Autonomous Landing Using An Mpc-controlled Micro UAV on A Moving Platform," in *International Conference on Unmanned Aircraft Systems (ICUAS)*, 2020, 771–780.
- J. Wubben, F. Fabra, C. T. Calafate, T. Krzeszowski, J. M. Marquez-barja, J.-c. Cano, and P. Manzoni, "Accurate Landing of Unmanned Aerial Vehicles Using Ground Pattern Recognition," *Electronics*, vol. 8, no. 12, Jan. 2019.
- J. Goslinski, W. Giernacki, and A. Krolikowski, "A Nonlinear Filter for Efficient Attitude Estimation of Unmanned Aerial Vehicle (UAV)," *Journal of Intelligent & Robotic Systems*, vol. 95, no. 3-4, 1079–1095, Jan. 2019.
- K. D. Nguyen and T. thang Nguyen, "Vision-based Software-in-the-loop-simulation for Unmanned Aerial Vehicles Using Gazebo and Px4 Open Source," in *International Conference on System Science And Engineering (ICSSE)*, 2019, 429–432.
- C. Potena, D. Nardi, and A. Pretto, "Joint Vision-based Navigation, Control and Obstacle Avoidance for UAVs In Dynamic Environments," in *European Conference on Mobile Robots (ECMR)*, 2019.

[3c] **V. Spurny**, T. Baca, M. Saska, R. Penicka, T. Krajnik, J. Thomas, D. Thakur, G. Loianno, and V. Kumar, "Cooperative Autonomous Search, Grasping, and Delivering in a Treasure Hunt Scenario by a Team of UAVs," *Journal of Field Robotics*, vol. 36, no. 1, 125–148, 2019

- C. Lenz, M. Schwarz, A. Rochow, J. Razlaw, A. S. Periyasamy, M. Schreiber, and S. Behnke, "Autonomous Wall Building with a UGV-UAV Team at MBZIRC 2020," in *International Symposium on Safety, Security, and Rescue Robotics (SSRR)*, 2020, 189–196.
- J. Goslinski, W. Giernacki, and A. Krolikowski, "A Nonlinear Filter for Efficient Attitude Estimation of Unmanned Aerial Vehicle (UAV)," *Journal of Intelligent & Robotic Systems*, vol. 95, no. 3-4, 1079–1095, Jan. 2019.
- W. Giernacki, "Iterative Learning Method for In-flight Auto-tuning of UAV Controllers Based on Basic Sensory Information," *Applied Sciences-basel*, vol. 9, no. 4, Jan. 2019.
- B. Lopez, J. Munoz, F. Quevedo, C. A. Monje, S. Garrido, and L. E. Moreno, "Path Planning and Collision Risk Management Strategy for Multi-UAV Systems In 3d Environments," *Sensors*, vol. 21, no. 13, Jan. 2021.
- M. Zhao, T. Anzai, F. Shi, T. Maki, T. Nishio, K. Ito, N. Kuromiya, K. Okada, and M. Inaba, "Versatile Multilinked Aerial Robot With Tilted Propellers: Design, Modeling, Control, and State Estimation for Autonomous Flight And Manipulation," *Journal of Field Robotics*,
- M. Basiri, J. Goncalves, J. Rosa, A. Vale, and P. Lima, "An Autonomous Mobile Manipulator To Build Outdoor Structures Consisting of Heterogeneous Brick Patterns," *Sn Applied Sciences*, vol. 3, no. 5, Jan. 2021.
- S. A. Emami and A. Banazadeh, "Simultaneous Trajectory Tracking and Aerial Manipulation Using A Multi-stage Model Predictive Control," *Aerospace Science and Technology*, vol. 112, Jan. 2021.

- C. J. Turner, R. Ma, J. Chen, and J. Oyekan, "Human In the Loop: Industry 4.0 Technologies and Scenarios for Worker Mediation of Automated Manufacturing," *IEEE Access*, vol. 9, 103950–103966, 2021.
- F. Real, A. R. Castano, A. Torres-gonzalez, J. Capitan, P. J. Sanchez-cuevas, M. J. Fernandez, M. Villar, and A. Ollero, "Experimental Evaluation of A Team of Multiple Unmanned Aerial Vehicles for Cooperative Construction," *IEEE Access*, vol. 9, 6817–6835, 2021.
- S. Rao, A. Chakravarthy, and D. Ghose, "Planar Manipulation of An Object By Unmanned Aerial Vehicles Using Sliding Modes," *Journal of Guidance Control and Dynamics*, vol. 44, no. 1, 120–137, Jan. 2021.
- G. Michieletto, A. Cenedese, L. Zaccarian, and A. Franchi, "Hierarchical Nonlinear Control for Multi-rotor Asymptotic Stabilization Based on Zero-moment Direction," *Automatica*, vol. 117, Jan. 2020.
- L. Campos-macias, R. Aldana-lopez, R. De La Guardia, J. I. Parra-vilchis, and D. Gomez-gutierrez, "Autonomous Navigation of Mavs In Unknown Cluttered Environments," *Journal of Field Robotics*, vol. 38, no. 2, 307–326, Jan. 2021.
- A. Mohiuddin, T. Tarek, Y. Zweiri, and D. Gan, "A Survey of Single and Multi-UAV Aerial Manipulation," *Unmanned Systems*, vol. 8, no. 2, 119–147, Jan. 2020.
- D. Sanalitro, H. J. Savino, M. Tognon, J. Cortes, and A. Franchi, "Full-pose Manipulation Control of A Cable-suspended Load With Multiple UAVs Under Uncertainties," *IEEE Robotics and Automation Letters*, vol. 5, no. 2, 2185–2191, Jan. 2020.
- B. E. Jackson, T. A. Howell, K. Shah, M. Schwager, and Z. Manchester, "Scalable Cooperative Transport of Cable-suspended Loads With UAVs Using Distributed Trajectory Optimization," *IEEE Robotics and Automation Letters*, vol. 5, no. 2, 3368–3374, Jan. 2020.
- M. Coppola, K. N. Mcguire, C. De Wagter, and G. C. H. E. De Croon, "A Survey on Swarming With Micro Air Vehicles: Fundamental Challenges And Constraints," *Frontiers In Robotics and AI*, vol. 7, Jan. 2020.
- H. Shen, X. Zhang, H. Lu, B. Tian, and Q. Zong, "State Estimation and Control for Micro Aerial Vehicles In Gps-denied Environments," in *12th International Conference on Wireless Communications and Signal Processing (WCSP)*, 2020, 1070–1075.
- M. Lopez Garcia and J. Martinez Carranza, "A First Cnn-based Approach Towards Autonomous Flight for Object Lifting," *Computacion Y Sistemas*, vol. 24, no. 3, 1219–1228, 2020.
- A. Mohiuddin, T. Taha, Y. Zweiri, and D. Gan, "UAV Payload Transportation Via RTDP Based Optimized Velocity Profiles," *Energies*, vol. 12, no. 16, Jan. 2019.
- F. Guo, M. Wei, M. Ye, J. Li, O. Mechali, and Y. Cao, "An Unmanned Aerial Vehicles Collaborative Searching and Tracking Scheme In Three-dimension Space," in *9th IEEE Annual International Conference on Cyber Technology In Automation, Control, and Intelligent Systems (IEEE-cyber)*, 2019, 1262–1266.

[22a] T. Roucek, M. Pecka, P. Cizek, T. Petracek, J. Bayer, V. Salansky, D. Hert, M. Petrlik, T. Baca, **V. Spurny**, *et al.*, "DARPA Subterranean Challenge: Multi-robotic exploration of underground environments," in *Modelling and Simulation for Autonomous Systems (MESAS)*, vol. 11995, Springer, 2019, pp. 274–290

- M. Basiri, J. Goncalves, J. Rosa, A. Vale, and P. Lima, "An autonomous mobile manipulator to build outdoor structures consisting of heterogeneous brick patterns," *Sn Applied Sciences*, vol. 3, no. 5, Jan. 2021.
- S. Yuan, H. Wang, and L. Xie, "Survey on Localization Systems and Algorithms for Unmanned Systems," *Unmanned Systems*, vol. 9, no. 2, 129–163, Jan. 2021.
- J. P. Queralta, J. Taipalmaa, B. C. Pullinen, V. K. Sarker, T. N. Gia, H. Tenhunen, M. Gabbouj, J. Raitoharju, and T. Westerlund, "Collaborative Multi-robot Search and Rescue: Planning, Coordination, Perception, and Active Vision," *IEEE Access*, vol. 8, 191617–191643, 2020.

[4c] V. Kratky, P. Petracek, **V. Spurny**, and M. Saska, "Autonomous Reflectance Transformation Imaging by a Team of Unmanned Aerial Vehicles," *IEEE Robotics and Automation Letters*, vol. 5, no. 2, pp. 2302–2309, 2020

- A. Jaspe-Villanueva, M. Ahsan, R. Pintus, A. Giachetti, F. Marton, and E. Gobbetti, "Web-based Exploration of Annotated Multi-Layered Relightable Image Models," *Acm Journal on Computing and Cultural Heritage*, vol. 14, no. 2, Si, Jan. 2021.
- A. Alcantara, J. Capitan, R. Cunha, and A. Ollero, "Optimal Trajectory Planning for Cinematography With Multiple Unmanned Aerial Vehicles," *Robotics and Autonomous Systems*, vol. 140, Jan. 2021.

[10a] T. Baca, M. Petrlik, M. Vrba, **V. Spurny**, R. Penicka, D. Hert, and M. Saska, "The MRS UAV System: Pushing the Frontiers of Reproducible Research, Real-world Deployment, and Education with Autonomous Unmanned Aerial Vehicles," *Journal of Intelligent & Robotic Systems*, vol. 102, no. 26, pp. 1–28, 1 2021

- G. Silano and L. Iannelli, "Mat-fly: An Educational Platform for Simulating Unmanned Aerial Vehicles Aimed To Detect and Track Moving Objects," *IEEE Access*, vol. 9, 39333–39343, 2021.
- F. Real, A. R. Castano, A. Torres-gonzalez, J. Capitan, P. J. Sanchez-cuevas, M. J. Fernandez, M. Villar, and A. Ollero, "Experimental Evaluation of A Team of Multiple Unmanned Aerial Vehicles for Cooperative Construction," *IEEE Access*, vol. 9, 6817–6835, 2021.

Bibliography

[1c] **V. Spurny**, V. Pritzl, V. Walter, M. Petrlik, T. Baca, P. Stepan, D. Zaitlik, and M. Saska, "Autonomous Firefighting Inside Buildings by an Unmanned Aerial Vehicle," *IEEE Access*, vol. 9, pp. 15 872–15 890, 2021.

[2c] G. Loianno, **V. Spurny**, J. Thomas, T. Baca, D. Thakur, D. Hert, R. Penicka, T. Krajnik, A. Zhou, A. Cho, M. Saska, and V. Kumar, "Localization, Grasping, and Transportation of Magnetic Objects by a team of MAVs in Challenging Desert-like Environments," *IEEE Robotics and Automation Letters*, vol. 3, no. 3, pp. 1576–1583, 2018.

[3c] **V. Spurny**, T. Baca, M. Saska, R. Penicka, T. Krajnik, J. Thomas, D. Thakur, G. Loianno, and V. Kumar, "Cooperative Autonomous Search, Grasping, and Delivering in a Treasure Hunt Scenario by a Team of UAVs," *Journal of Field Robotics*, vol. 36, no. 1, 125–148, 2019.

[4c] V. Kratky, P. Petracek, **V. Spurny**, and M. Saska, "Autonomous Reflectance Transformation Imaging by a Team of Unmanned Aerial Vehicles," *IEEE Robotics and Automation Letters*, vol. 5, no. 2, pp. 2302–2309, 2020.

[5c] **V. Spurny**, M. Petrlik, V. Vonasek, and M. Saska, "Cooperative Transport of Large Objects by a Pair of Unmanned Aerial Systems using Sampling-based Motion Planning," in *IEEE International Conference on Emerging Technologies and Factory Automation (ETFA)*, 2019, pp. 955–962.

[6c] M. Saska, V. Kratky, **V. Spurny**, and T. Baca, "Documentation of Dark Areas of Large Historical Buildings by a Formation of Unmanned Aerial Vehicles using Model Predictive Control," in *IEEE International Conference on Emerging Technologies and Factory Automation (ETFA)*, 2017, pp. 1–8.

[7a] M. Saska, T. Krajnik, V. Vonasek, Z. Kasl, **V. Spurny**, and L. Preucil, "Fault-Tolerant Formation Driving Mechanism Designed for Heterogeneous MAVs-UGVs Groups," *Journal of Intelligent and Robotic Systems*, vol. 73, no. 1-4, pp. 603–622, 2014.

[8a] M. Saska, **V. Spurny**, and V. Vonasek, "Predictive control and stabilization of nonholonomic formations with integrated spline-path planning," *Robotics and Autonomous Systems*, vol. 75, pp. 379–397, 2016.

[9a] T. Baca, P. Stepan, **V. Spurny**, D. Hert, R. Penicka, M. Saska, J. Thomas, G. Loianno, and V. Kumar, "Autonomous Landing on a Moving Vehicle with an Unmanned Aerial Vehicle," *Journal of Field Robotics*, vol. 36, pp. 874–891, 5 2019.

[10a] T. Baca, M. Petrlik, M. Vrba, **V. Spurny**, R. Penicka, D. Hert, and M. Saska, "The MRS UAV System: Pushing the Frontiers of Reproducible Research, Real-world Deployment, and Education with Autonomous Unmanned Aerial Vehicles," *Journal of Intelligent & Robotic Systems*, vol. 102, no. 26, pp. 1–28, 1 2021.

[11a] D. Horla, W. Giernacki, T. Baca, **V. Spurny**, and M. Saska, "AL-TUNE: In-flight tuning of an altitude controller for UAVs," *Journal of Intelligent & Robotic Systems*, vol. 103, no. 5, pp. 1–16, 1 2021.

[12a] M. Vrba, Y. Stasinchuk, T. Baca, **V. Spurny**, M. Petrlik, D. Hert, D. Zaitlik, and M. Saska, "Autonomous Capturing of Agile Flying Objects using UAVs: The MBZIRC 2020 Challenge," Aug. 2021, Submitted to *Robotics and Autonomous Systems*.

[13a] T. Baca, R. Penicka, P. Stepan, M. Petrlik, **V. Spurny**, D. Hert, and M. Saska, "Autonomous Cooperative Wall Building by a Team of Unmanned Aerial Vehicles in the MBZIRC 2020 Competition," Dec. 2020, Submitted to *Robotics and Autonomous Systems*.

The core publications of this thesis are referenced with [*c] and the other author's publications with [*a].

[14a] V. Walter, **V. Spurny**, M. Petrlik, T. Baca, D. Zaitlik, L. Demkiv, and M. Saska, "Extinguishing of Real Fires by Fully Autonomous MAVs in the MBZIRC 2020 Competition," May 2021, Accepted to *Field Robotics*.

[15a] Y. Stasinchuk, M. Vrba, T. Baca, **V. Spurny**, M. Petrlik, D. Hert, D.Zaitlik, and M.Saska, "A Multi-MAV System For Autonomous Multiple Targets Elimination in the MBZIRC 2020 competition," May 2021, Accepted to *Field Robotics*.

[16a] K. Jindal, A. Wang, D. Thakur, A. Zhou, **V.Spurny**, V.Walter, G. Broughton, T. Krajnik, M. Saska, and G. Loianno, "Design and Deployment of an Autonomous Unmanned Ground Vehicle for Urban Firefighting Scenarios," Apr. 2021, Accepted to *Field Robotics*.

[17a] T. Roucek, M. Pecka, P. Cizek, T. Petricek, J. Bayer, V. Salansky, D. Hert, M. Petrlik, T. Baca, **V. Spurny**, V. Kratky, P. Petracek, D. Baril, M. Vaidis, V. Kubelka, F. Pomerleau, J. Faigl, K. Zimmermann, M. Saska, T. Svoboda, and T. Krajnik, "System for multi-robotic exploration of underground environments CTU-CRAS-NORLAB in the DARPA Subterranean Challenge," Jun. 2021, Accepted to *Field Robotics*.

[18a] V. Walter, **V. Spurny**, M. Petrlik, T. Baca, D. Zaitlik, and M. Saska, "Extinguishing of Ground Fires by Fully Autonomous UAVs Motivated by the MBZIRC 2020 Competition," in *International Conference on Unmanned Aircraft Systems (ICUAS)*, 2021, pp. 787–793.

[19a] M. Saska, **V. Spurny**, and L. Preucil, "Trajectory Planning and Stabilization for Formations Acting in Dynamic Environments," in *Progress in Artificial Intelligence*, Springer, 2013, pp. 319–330.

[20a] **V. Spurny**, T. Baca, and M. Saska, "Complex manoeuvres of heterogeneous MAV-UGV formations using a model predictive control," in *International Conference on Methods and Models in Automation and Robotics (MMAR)*, 2016, pp. 998–1003.

[21a] M. Vrba, J. Pogran, V. Pritzl, **V. Spurny**, and M. Saska, "Real-time localization of transmission sources using a formation of micro aerial vehicles," in *IEEE International Conference on Real-time Computing and Robotics (RCAR)*, 2019, pp. 203–208.

[22a] T. Roucek, M. Pecka, P. Cizek, T. Petracek, J. Bayer, V. Salansky, D. Hert, M. Petrlik, T. Baca, **V. Spurny**, F. Pomerleau, V. Kubelka, J. Faigl, K. Zimmermann, M. Saska, T. Svoboda, and J. Krajnik, "DARPA Subterranean Challenge: Multi-robotic exploration of underground environments," in *Modelling and Simulation for Autonomous Systems (MESAS)*, vol. 11995, Springer, 2019, pp. 274–290.

[23a] J. Faigl, P. Vana, M. Saska, T. Baca, and **V. Spurny**, "On solution of the Dubins touring problem," in *IEEE European Conference on Mobile Robots (ECMR)*, 2017, pp. 1–6.

[24a] Y. Stasinchuk, M. Vrba, M. Petrlik, T. Baca, **V. Spurny**, D. Hert, D. Zaitlik, T. Nascimento, and M. Saska, "A Multi-UAV System for Detection and Elimination of Multiple Targets," Accepted to *IEEE International Conference on Robotics and Automation (ICRA)*, Jun. 2021.

[25] H. Shakhatreh, A. H. Sawalmeh, A. Al-Fuqaha, Z. Dou, E. Almaita, I. Khalil, N. S. Othman, A. Khreishah, and M. Guizani, "Unmanned Aerial Vehicles (UAVs): A Survey on Civil Applications and Key Research Challenges," *IEEE Access*, vol. 7, pp. 48 572–48 634, 2019.

[26] S. Verykokou, C. Ioannidis, G. Athanasiou, N. Doulamis, and A. Amditis, "3D reconstruction of disaster scenes for urban search and rescue," *Multimedia Tools and Applications*, vol. 77, no. 8, pp. 9691–9717, 2018.

[27] D. Duarte, F. Nex, N. Kerle, and G. Vosselman, "Towards a more efficient detection of earthquake induced facade damages using oblique uav imagery," *IEEE International Conference on Unmanned Aerial Vehicles and Geomatics (ICUAVG)*, pp. 93–100, 2017.

The core publications of this thesis are referenced with [*c] and the other author's publications with [*a].

[28] Y. Yamazaki, M. Tamaki, C. Premachandra, C. J. Perera, S. Sumathipala, and B. H. Sudantha, "Victim Detection Using UAV with On-board Voice Recognition System," in *IEEE International Conference on Robotic Computing (IRC)*, 2019, pp. 555–559.

[29] M. Rahnemoonfar, R. Murphy, M. V. Miquel, *et al.*, "Flooded Area Detection from UAV Images Based on Densely Connected Recurrent Neural Networks," in *International Geoscience and Remote Sensing Symposium (IGARSS)*, 2018, pp. 1788–1791.

[30] R. Ravichandran, D. Ghose, and K. Das, "UAV Based Survivor Search during Floods," in *International Conference on Unmanned Aircraft Systems (ICUAS)*, 2019, pp. 1407–1415.

[31] C. Yuan *et al.*, "UAV-based forest fire detection and tracking using image processing techniques," in *International Conference on Unmanned Aircraft Systems (ICUAS)*, 2015, pp. 639–643.

[32] F. Real, R. Castaño, A. Torres-González, J. Capitán, P. J. Sánchez-Cuevas, M. J. Fernández, M. Villar, and A. Ollero, "Experimental Evaluation of a Team of Multiple Unmanned Aerial Vehicles for Cooperative Construction," *IEEE Access*, vol. 9, pp. 6817–6835, 2021.

[33] *DARPA Subterranean (SubT) Challenge*, Accessed 2021-11-01
[Online]. Available: `https://www.subtchallenge.com/`.

[34] *The Mohamed Bin Zayed International Robotics Challenge (MBZIRC)*, Accessed 2021-11-01
[Online]. Available: `https://www.mbzirc.com`.

[35] *Wing*, Accessed 2021-11-01
[Online]. Available: `https://wing.com/`.

[36] *Amazon Prime Air*, Accessed 2021-11-01
[Online]. Available: `https://www.amazon.com/Amazon-Prime-Air/b?ie=UTF8&node=8037720011`.

[37] V. Chamola, V. Hassija, V. Gupta, and M. Guizani, "A Comprehensive Review of the COVID-19 Pandemic and the Role of IoT, Drones, AI, Blockchain, and 5G in Managing its Impact," *IEEE Access*, vol. 8, pp. 90 225–90 265, 2020.

[38] E. Umili, M. Tognon, D. Sanalitro, G. Oriolo, and A. Franchi, "Communication-based and Communication-less approaches for Robust Cooperative Planning in Construction with a Team of UAVs," in *International Conference on Unmanned Aircraft Systems (ICUAS)*, 2020, pp. 279–288.

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

[40] P. Petráček, V. Walter, T. Báča, and M. Saska, "Bio-Inspired Compact Swarms of Unmanned Aerial Vehicles without Communication and External Localization," *Bioinspiration & Biomimetics*, vol. 16, no. 2, p. 026 009, 2020.

[41] M. Coppola, K. N. McGuire, C. De Wagter, and G. C. de Croon, "A Survey on Swarming With Micro Air Vehicles: Fundamental Challenges and Constraints," *Frontiers in Robotics and AI*, vol. 7, p. 18, 2020.

[42] L. Merino, F. Caballero, J. R. Martínez-De-Dios, I. Maza, and A. Ollero, "An unmanned aircraft system for automatic forest fire monitoring and measurement," *Journal of Intelligent & Robotic Systems*, vol. 65, no. 1-4, pp. 533–548, 2012.

[43] A. Restas, "Forest fire management supporting by UAV based air reconnaissance results of Szendro fire department, Hungary," in *International Symposium on Environment Identities and Mediterranean Area (ISEIMA)*, 2006, pp. 73–77.

[44] A. Viguria, I. Maza, and A. Ollero, "Distributed Service-Based Cooperation in Aerial/Ground Robot Teams Applied to Fire Detection and Extinguishing Missions," *Advanced Robotics*, vol. 24, no. 1-2, pp. 1–23, 2010.

[45] R. Chen, H. Cao, H. Cheng, and J. Xie, "Study on Urban Emergency Firefighting Flying Robots Based on UAV," *IEEE Advanced Information Technology, Electronic and Automation Control Conference (IAEAC)*, pp. 1890–1893, 2019.

[46] H. Qin, J. Q. Cui, J. Li, Y. Bi, M. Lan, M. Shan, W. Liu, K. Wang, F. Lin, Y. Zhang, *et al.*, "Design and implementation of an unmanned aerial vehicle for autonomous firefighting missions," in *IEEE International Conference on Control & Automation (ICCA)*, 2016, pp. 62–67.

[47] P. Pecho, P. Magdolenová, and M. Bugaj, "Unmanned aerial vehicle technology in the process of early fire localization of buildings," *Transportation Research Procedia*, International Scientific Conference on Sustainable, Modern and Safe Transport (TRANSCOM), vol. 40, pp. 461–468, 2019.

[48] A. Imdoukh, A. Shaker, A. Al-Toukhy, D. Kablaoui, and M. El-Abd, "Semi-autonomous indoor firefighting UAV," in *International Conference on Advanced Robotics (ICAR)*, 2017, pp. 310–315.

[49] C. Lenz, M. Schwarz, A. Rochow, J. Razlaw, A. S. Periyasamy, M. Schreiber, and S. Behnke, "Autonomous Wall Building with a UGV-UAV Team at MBZIRC 2020," in *IEEE International Symposium on Safety, Security, and Rescue Robotics (SSRR)*, 2020, pp. 189–196.

[50] D. Mellinger, M. Shomin, N. Michael, and V. Kumar, "Cooperative grasping and transport using multiple quadrotors," in *Distributed autonomous robotic systems*, Springer, 2013, pp. 545–558.

[51] R. Ritz and R. D'Andrea, "Carrying a flexible payload with multiple flying vehicles," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2013, pp. 3465–3471.

[52] Z. Wang, S. Singh, M. Pavone, and M. Schwager, "Cooperative Object Transport in 3D with Multiple Quadrotors Using No Peer Communication," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2018, pp. 1064–1071.

[53] P. O. Pereira, P. Roque, and D. V. Dimarogonas, "Asymmetric Collaborative Bar Stabilization Tethered to Two Heterogeneous Aerial Vehicles," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2018, pp. 5247–5253.

[54] H. Lee, H. Kim, W. Kim, and H. J. Kim, "An Integrated Framework for Cooperative Aerial Manipulators in Unknown Environments," *IEEE Robotics and Automation Letters*, vol. 3, no. 3, pp. 2307–2314, 2018.

[55] V. Vonásek, J. Faigl, T. Krajník, and L. Přeučil, "RRT-path–a guided rapidly exploring random tree," in *Robot Motion and Control*, Springer, 2009, pp. 307–316.

[56] Y. Zhang and K.-L. Ma, "Lighting design for globally illuminated volume rendering," *IEEE Transactions on Visualization and Computer Graphics*, vol. 19, no. 12, pp. 2946–2955, 2013.

[57] J. Birn, *Digital lighting & rendering*. Pearson Education, 2014.

[58] R. Sheppard, *Digital photography simplified*. John Wiley & Sons, 2008, vol. 9.

[59] H. Michalska and D. Q. Mayne, "Robust receding horizon control of constrained nonlinear systems," *IEEE transactions on automatic control*, vol. 38, no. 11, pp. 1623–1633, 1993.

[60] H. Mytum and J. Peterson, "The application of Reflectance Transformation Imaging (RTI) in historical archaeology," *Historical Archaeology*, vol. 52, no. 2, pp. 489–503, 2018.

[61] S. N. Manrique Tamayo, J. C. Valcárcel Andrés, and M. Osca Pons, "Applications of reflectance transformation imaging for documentation and surface analysis in conservation," *International Journal of Conservation Science*, vol. 4, pp. 535–548, 2013.

[62] K. A. Ghamry and Y. Zhang, "Formation control of multiple quadrotors based on leader-follower method," in *International Conference on Unmanned Aircraft Systems (ICUAS)*, 2015, pp. 1037–1042.

[63] B. Yun, B. M. Chen, K. Y. Lum, and T. H. Lee, "Design and implementation of a leader-follower cooperative control system for unmanned helicopters," *Journal of Control Theory and Applications*, vol. 8, no. 1, pp. 61–68, 2010.

[64] M. A. Kamel, K. A. Ghamry, and Y. Zhang, "Real-time fault-tolerant cooperative control of multiple UAVs-UGVs in the presence of actuator faults," in *International Conference on Unmanned Aircraft Systems (ICUAS)*, 2016, pp. 1267–1272.

[65] J. Choi and Y. Kim, "Fuel efficient three dimensional controller for leader-follower UAV formation flight," in *International Conference on Control, Automation and Systems (ICCAS)*, 2007, pp. 806–811.

[66] M. Quigley, K. Conley, B. Gerkey, J. Faust, T. Foote, J. Leibs, R. Wheeler, and A. Y. Ng, "ROS: an open-source Robot Operating System," in *ICRA workshop on open source software*, Kobe, Japan, vol. 3, 2009, p. 5.

[67] N. Koenig and A. Howard, "Design and use paradigms for gazebo, an open-source multi-robot simulator," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, vol. 3, 2004, pp. 2149–2154.

[68] L. Meier, D. Honegger, and M. Pollefeys, "PX4: A node-based multithreaded open source robotics framework for deeply embedded platforms," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2015, pp. 6235–6240.

[69] M. Vrba, D. Heřt, and M. Saska, "Onboard Marker-Less Detection and Localization of Non-Cooperating Drones for Their Safe Interception by an Autonomous Aerial System," *IEEE Robotics and Automation Letters*, vol. 4, no. 4, pp. 3402–3409, 2019.

[70] V. Walter, M. Saska, and A. Franchi, "Fast Mutual Relative Localization of UAVs using Ultraviolet LED Markers," in *International Conference on Unmanned Aircraft Systems (ICUAS)*, 2018, pp. 1217–1226.

[71] V. Walter, N. Staub, M. Saska, and A. Franchi, "Mutual Localization of UAVs based on Blinking Ultraviolet Markers and 3D Time-Position Hough Transform," in *IEEE 14th International Conference on Automation Science and Engineering (CASE)*, 2018, pp. 298–303.

[72] V. Walter, N. Staub, A. Franchi, and M. Saska, "Uvdar system for visual relative localization with application to leader–follower formations of multirotor uavs," *IEEE Robotics and Automation Letters*, vol. 4, no. 3, pp. 2637–2644, 2019.

[73] T. Baca, M. Jilek, P. Manek, P. Stibinger, V. Linhart, J. Jakubek, and M. Saska, "Timepix Radiation Detector for Autonomous Radiation Localization and Mapping by Micro Unmanned Vehicles," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2019, pp. 1129–1136.

[74] T. Baca, D. Turecek, R. McEntaffer, and R. Filgas, "Rospix: modular software tool for automated data acquisitions of Timepix detectors on Robot Operating System," *Journal of Instrumentation*, vol. 13, no. 11, p. C11008, 2018.

[75] P. Štibinger, T. Báča, and M. Saska, "Localization of Ionizing Radiation Sources by Cooperating Micro Aerial Vehicles With Pixel Detectors in Real-Time," *IEEE Robotics and Automation Letters*, vol. 5, no. 2, pp. 3634–3641, 2020.

[76] R. Pěnička, J. Faigl, M. Saska, and P. Váňa, "Data collection planning with non-zero sensing distance for a budget and curvature constrained unmanned aerial vehicle," *Autonomous Robots*, vol. 43, no. 8, pp. 1937–1956, 2019.

[77] R. Pěnička, J. Faigl, and M. Saska, "Physical orienteering problem for unmanned aerial vehicle data collection planning in environments with obstacles," *IEEE Robotics and Automation Letters*, vol. 4, no. 3, pp. 3005–3012, 2019.

[78] R. Pěnička, J. Faigl, P. Váňa, and M. Saska, "Dubins orienteering problem," *IEEE Robotics and Automation Letters*, vol. 2, no. 2, pp. 1210–1217, 2017.

[79] M. Saska, "Large sensors with adaptive shape realised by self-stabilised compact groups of micro aerial vehicles," in *Robotics Research*, Springer International Publishing, 2020, pp. 101–107.

[80] A. Ahmad, V. Walter, P. Petracek, M. Petrlik, T. Baca, D. Zaitlik, and M. Saska, "Autonomous aerial swarming in gnss-denied environments with high obstacle density," Accepted to *IEEE International Conference on Robotics and Automation (ICRA)*, 2021.

[81] M. Vrba and M. Saska, "Marker-Less Micro Aerial Vehicle Detection and Localization Using Convolutional Neural Networks," *IEEE Robotics and Automation Letters*, vol. 5, no. 2, pp. 2459–2466, 2020.

[82] R. Bähnemann, D. Schindler, M. Kamel, R. Siegwart, and J. Nieto, "A decentralized multi-agent unmanned aerial system to search, pick up, and relocate objects," in *IEEE International Symposium on Safety, Security and Rescue Robotics (SSRR)*, IEEE, 2017, pp. 123–128.

[83] M. Nieuwenhuisen, M. Beul, R. A. Rosu, J. Quenzel, D. Pavlichenko, S. Houben, and S. Behnke, "Collaborative object picking and delivery with a team of micro aerial vehicles at MBZIRC," in *European Conference on Mobile Robots (ECMR)*, IEEE, 2017, pp. 1–6.

[84] A. L. Majdik, L. Tizedes, M. Bartus, and T. Szirányi, "Photogrammetric 3D reconstruction of the old slaughterhouse in Budapest," in *International Workshop on Computational Intelligence for Multimedia Understanding (IWCIM)*, IEEE, 2016, pp. 1–5.