**Autonomous mapping with Quad-rotor UAVs**

Abstract: The goal is to implement software for creating a map using a vision-based quadrotor UAV with a focus on an indoor use-case. The first step is to implement the map computation, followed by a planning algorithm to explore unknown regions on the map. The final step is coordination among multiple UAVs to explore collaboratively.

In March 2011, Japan suffered from its biggest earthquake and most devastating tsunami. Severe damage was inflicted on its Fukushima nuclear plants and more than 100,000 people were evacuated after radi- ation levels became unsafe. Rescue workers could not operate on-site; this prevented them from securing safety at the atomic power plant and caused a major leak. One month after the disaster, a small aerial vehicle equipped with cameras was sent to take pic- tures and videos of the affected areas. The video footages obtained brought valuable information to the rescue teams that could not have been acquired oth- erwise. Nonetheless, the use of aerial vehicles is still severely limited because they must be controlled by a remote operator within transmission range. It is also necessary to have an operator to control the camera and interpret the data. An emerging promising solu- tion is autonomous aerial vehicles.

In order to work autonomously, these systems need to be intelligent and rational. They need high levels of knowledge to accomplish their complex missions. They have to adapt to unexpected situations such as changes in the environment and possible loss of sensor data, e.g., GPS due to obstructing buildings or indoor exploration. In a multi-UAV setting, they additionally need to communicate with each other to delegate tasks and learn from the information acquired by other UAVs. Every mission is unique in terms of deployment areas and goals. Since missions can be safety critical, in that human lives may be at stake, the correctness of implementations must be formally verified.

A famous example of an implementation error is a numeric overflow that caused Ariane 5 to self-destruct immediately after take-off. Another example from 1996 is the Lockheed Martin/Boeing Darkstar long-endurance UAV, which crashed following what the Pentagon called a “mishap [. . . ] directly traceable to deficiencies in the modelling and simulation of the flight vehicle”.

Manual inspection of complex software is error-prone and costly, so tool support is in dire need. Although tools exist to uncover errors in standard software, no mechanism to certify and formally verify the behaviour and implementation of UAVs with respect to high-level rules and requirements is available. A further challenge in developing such mechanisms is the concurrency inherent in all aspects of multi-UAVs.

The method of choice in formal verification is specifying behavioural properties in a logic and proving correctness using automated deduction. To quote: “Logic has turned out to be significantly more effective in computer science in the past 30 years than it has been in mathematics in the past 100 years, insomuch that it is called the calculus of computer science. [...] The extensive and continuous connections between logic and computer science are a matter of engineering practice at every level of the field from artificial intelligence to software engineering.”

In this project we will use logics of information flow and certifiable tools to prove that multi-UAVs learn correctly from each other and their environment. We will develop a formal link between the low-level code that runs on the embedded CPUs and the high-level logical abstraction.

We use search and rescue multi-UAVs as the experimental platform. Because of their specific kinematic and safety constraints, existing sets of rules are insufficient. We aim to design scenarios and high-level rules for multi-UAV navigation, communication and safety. We consider two classes of search scenarios: outdoor and indoor exploration. The overall mission goal is to explore as much as possible of a given search area as quickly as possible. When outdoors, UAVs must explore an open area (e.g., a park). They use GPS to determine their position with a certain degree of accuracy. When indoors, UAVs must explore a building with one level and multiple rooms. The constraints here include unavailability of GPS positioning. Instead, inertial data and camera measurements as well as priors on the environment will be used for on-board pose estimation and mapping in real-time. The rules formalising the mission goal include: 1) plan motion according to prior map, 2) if two or more UAVs have the same prior information, then co- ordinate the exploration step, and 3) avoid collision. The coordination step can be further broken down into a) if one UAV performs an action, the other should not do the same, and b) if a UAV hears another UAV, it should broadcast its map.

Dr Daniel Kroening, Computer Science