Plan B report: autonomous navigation and vision-based landing on a moving platform

Roman Ripp
Department of Computer Science
University of Minnesota
rippx009@umn.edu

I. INTRODUCTION

In this report we introduce design of a control framework for Vertical Take-off and Landing Unmanned Aerial Vehicle (VTOL UAV). The goal of the project is to provide framework for collaborative use of UAV and UGV [10]. The motivation is inspired by limitations of commercially available ground and aerial robots. Short battery life of low-cost VTOL UAVs makes them unusable in cases where long distance flights are needed. Inability to navigate through harsh terrain puts limitations on UGVs. An example of such mission can be crop inspection for precision agriculture. The narrow passages between crops restrict access for UGVs, the substantial area needed to be covered will require hours of flight time or multiple UAVs. This approach can be time and resource consuming.

In this report we describe system where UAV is deployed on demand from UGV, and after returned back [10]. Such system combines long operation time of UGV with versatility of UAV.

Many papers were published on autonomous navigation and landing [9], [3], [8] and even production grade solutions exist. For instance Lockheed Martin X-47B is capable of landing on an aircraft carrier. Unlike available systems we are concerned with overall cost, hence we forbear from using high precision position estimation equipment. In this project we use commercially available electronics and off-the-shelf robotics platforms.

II. SYSTEM DESCRIPTION

The system consists of a Ground Station (GS), Unmanned Aerial Vehicle (UAV) and Unmanned Ground Vehicle (UGV). The GS is a computer running control and planning software. It is equipped with a video capturing device PS3 Playstation Eye Camera. The communication between GS and UAV is provided with wireless communication device XBee PRO 900HP. As UAV in this project we used vertical take-off and landing aerial vehicle MikroKopter HexaXL [4]. As a ground vehicle we used Clearpath Husky A200 [2]. The GS can be mounted on a UGV.

GS is used for high level control and coordination between UGV and UAV. The main functionality of a GS includes position estimation of UAV, behavior control and coordination between UAV and UGV. Additional functionality provides flight parameters logging and communication link maintenance. The low-level control of UAV, such as motor thrust balancing, is executed onboard by ATMEGA1284P microcontroller. Low-level control of UGV is executed by GS. The principal scheme is presented on a figure 1.

A. UAV Driver

Software organization of a system is presented on figure 2. UAV Driver is medium between high-level control in ROS and MikroKopter firmware on board. This module receives control from UAV Behavior controller, interprets message to MikroKopter serial communication format. The message is encrypted and send through serial communication link. Driver manages communication link state and initiates connection recovery then needed. UAV Driver queries UAV for telemetry data at a rate of 2.5hz. The rate is adjustable however, lower frequency recommended for less bandwidth use. Telemetry data is published as a ROS topic and can be logged or used at higher levels of control. The data consists of 35 flight parameters such as battery charge, current location and altitude, etc. Figure 3 presents recorded data of an autonomous flight.

B. Behavior control

MikroKopter HexaXL provides operator with following functionality: GPS position hold, altitude hold, waypoint nav-
igation, direct control through wireless link. More information on standard functionality can be found at [4]. In this project we extend functionality of MikroKopter by implementing autonomous takeoff, landing and navigation behaviors, we also provide a method for switch between behaviors without interposition from operator. We provide interfaces compatible with Robot Operating System (ROS).

GS interfaces onboard controller through RF link and initiates required routines that are executed locally on UGV. One of the advantages of current approach is versatility. Developer can implement trajectory planning, multi-robot collaboration or other high-level algorithms without worrying about low-level control such as UGV dynamics and stabilization. We provide proof of concept in section IV.

Three main behavior routines were implemented: takeoff, landing and navigation. First experimental results II-B revealed flaws of the system. Switching between behaviors can introduce unbounded delays due to communication, control commands have to reach MikroKopter UAV in strict order. Control messages are often lost. These undermined system robustness leaving UAV in ambiguous state. We have improved the system by implementing message verification, communication timeouts and "stay-still-then-switch" approach. GS is monitoring the condition of communication link and if it is unavailable takes measures to reestablish connection. The monitoring process does not consume additional bandwidth because it uses telemetry messages from UAV for quality estimation. GS verifies every control message and if no acknowledgement is received resends the message after timeout period. The next command is not sent before previous is verified. The method described above increases robustness of a system at cost of delay. The overhead due to message verification was examined and result presented in experiments section II-B.

Then in switching state the UAV is not controlled by GS, nor operator. Given that delays can be order of seconds further improvements has been made. To eliminate ambiguity in switching state before every behavior switch UAV is set to position hold mode. The graphical representation can be seen on figure 4. When position hold mode is initiated and no control signals is received the UAV maintains its current position and altitude. UAV will hover in place until communication issues resolved by the GS.

MikroKopter can be controlled with direct control commands such as throttle, roll, pitch, yaw values. For efficiency purposes we did not implement verification for direct control commands. Nevertheless time between GS issuing command and UAV executing is sufficient and increases with distance between UGV and GS. Therefore, use of direct control is possible only in conjunction with dynamic position hold and within close proximity to GS.

Takeoff behavior is a routine that uses the PD controller to adjust thrust and maintain desired altitude. In takeoff position of the UAV relative to the starting point is maintained using...
dynamic position hold. Once desired altitude is reached GS hands over control to onboard controller.

Waypoint navigation behavior includes waypoint generation phase that takes place on GS. After, waypoints are transmitted to the UAV. Finally, The navigation is triggered by a GS.

Autonomous landing behavior consists of three stages. First, UAV and UGV agree on common meeting point. In section II-C we provide algorithm details. Second, navigation towards meeting point takes place. This is not different from waypoint navigation described in previous paragraph. Third phase is the vision-aided landing. The UAV is guided by a base station until it is landed at desired location. The description of control algorithm used is presented in section II-E.

C. Ground and aerial robot coordination

The goal of this stage is to derive common location, where UAV and UGV meet. Here and after we will refer to such location as rendezvous point. The following conditions for rendezvous should be satisfied: UAV should be in the field of view of the camera mounted on UGV, trajectory of UGV is constant and consists of set of straight lines, both robots arrive to rendezvous location at the same time.

On coordination initialization we collect the following information: current geographical coordinates of UAV and UGV, list of waypoints for UGV. Using this data rendezvous point can be derived. The setting is presented on figure 5. The calculation of rendezvous point in geographic coordinates requires substantial computation and use of numeric solving methods. Therefore, for simplicity we convert position data from geographical coordinates to cartesian coordinate frame. Derive rendezvous point and convert back to geographic coordinate representation. We start with finding rendezvous location in horizontal plane. Solving system of equations 1, with respect to x_r, y_r will yield the coordinates of rendezvous point.

\[
\begin{align*}
\max_n \left( \sum_{i=1}^{n} d_i \leq D_{UGV} \right) \\
\frac{n \cdot t_{turn} + D_{UGV}}{V_{UGV}} = \frac{D_{UAV}}{V_{UAV}} \\
\frac{(y_n - y_n)^2}{(x_n - x_n)^2} \\
D_{UAV} = \sqrt{(x_a - x_r)^2 + (y_a - y_r)^2} \\
D_{UGV} = d_r + \sum_{i=1}^{n} d_i \\
d_r = \sqrt{(x_r - x_n)^2 + (y_r - y_n)^2} \\
d_i = \sqrt{(x_i - x_{i-1})^2 + (y_i - y_{i-1})^2}
\end{align*}
\]

Here n - is an number of waypoints on UGV path before rendezvous point. D_{UGV} and D_{UAV} are the total distances form initial robot positions to the rendezvous point. t_{turn} is time UGV spends at each waypoint before proceeding on the path. V_{UAV} and V_{UGV} - average ground velocities of UAV and UGV.

Once we have rendezvous coordinates we compute approach altitude for UAV. The goal is to guarantee that UAV will be inside the view cone of the camera 6. Altitude is derived with equation 6.

\[
\begin{align*}
z = (e_{UGV} + e_{UAV}) \cos(a) + h \\
z < d
\end{align*}
\]

Altitude depends on accuracy of the position estimation unit of both UAV and UGV, e_{UGV} and e_{UAV}, elevation of UGV h, angle of camera view a, and maximum detection distance d for given AR marker.

D. Vision-based position estimation

One of aspects of autonomy in this project is ability to land on a UGV moving on a defined path. This operation requires accurate position and orientation estimation for UAV with respect to UGV. We propose vision-based approach for the task.

Vision system consists of a mono web camera connected to a GS. An Augmented Reality (AR) marker is attached to
UAV body. As AR marker we have chosen an image pattern printed on a flat surface. The requirements for the marker are as follows: marker must have square outer boundary, have continuous border, image must not be rotationally symmetrical. Figure 7 presents an example of AR marker used in this project. The size, color and marker image can change. Experimental results described in section IV demonstrate how adjusting parameters of a marker increase quality of position estimation.

We used algorithms from OpenCV library and ARToolKit to filter the video stream, detect presence of the AR marker and derive relative position of the marker with respect to the camera. The position of UAV can be derived since marker is attached to UAV.

Marker detection and tracking is a corner detection fast pose estimation algorithm provided by ARToolKit library [7]. Position estimation is an eight step process performed on each video frame. Steps of a process presented on figure 8. We start with image is thresholding, this simplest image segmentation method converts frame into bi-color map. Next, connected components are detected and contours are derived. Later corners of the contoured objects are detected and normalized. Normalization process reconstructs the image back to square frame. The resulting image is matched against the marker pattern and if similarity is sufficient the transformation between camera and marker is computed. The result is position and orientation of a marker in the cartesian coordinate frame with camera being an origin.

E. Vision-aided landing

The final step of landing procedure is vision-aided tracking and descend. The landing controller uses the position estimate computed as described in section II-D to guide UAV until it is set on top of UGV.

We have used two types of controllers for the task simple PID [5] and backstepping [1]. We are minimizing the error between desired position and actual position of the UAV by controlling roll, pitch, yaw and throttle values. The altitude remains constant until position error in horizontal plane is greater than desired precision. If precision is achieved UAV starts to descend.

III. SIMULATION RESULTS

The objective of simulation lay in UAV controller tuning. We want to increase robustness of the system without risk of damaging the UAV. We have tested autonomous landing procedure using SwarmSimX environment [6]. The setting included UGV following straight line trajectory and UAV tracking it and descending than possible.

We implemented physical model of a hexacopter and simulated its control interface. This allow us to tune control algorithms in simulator and switch to real hardware easily.

The simulation environment initially has ideal conditions meaning communication is perfect and no disturbing forces are affecting UAV. This conditions are rarely met by a real robot, hence test cases to verify robustness were implemented. We simulate presence of delay in communication, and disturbance due to air flows. Random force vectors applied to the frame of the UAV. The results show that system can withstand short term forces at cost of landing time. The maximum acceptable communication delay was measured at 245 ms. The time needed for landing without communication delay in average is 43 seconds, given the initial altitude of 3 meters and horizontal plane error 8.5 meters.

IV. EXPERIMENTAL RESULTS

A. Vision

For vision system test of performance we used rotationally asymmetric black and white pattern 210 mm, figure in size 7. Calibrated ps3 Playstation Eye camera with autofocus.

The ability to detect and track AR marker was tested both in indoor and outdoor setting. Based on observations maximum distance between marker and camera where detection is possible is 3 meters for 210 mm marker. The effective distance with detection confidence above 0.8 is 1.42 m. Average error was less than 13 mm with 1.42 m and 134 mm with 3.0 m marker-camera distance.

It was determined that level of illumination is affecting the detection confidence. It is possible to increase the detection by training AR marker with current illumination conditions. We further increased the robustness of the marker detection by reducing the reflective qualities of the marker surface. The improved vision system has effective distance up to 2.1 m and maximum distance 3.2 m.
B. Autonomous navigation

The ability to perform autonomous flight was tested. The experiment setting included set of four waypoints, and the coordinates of the landing location. The goal of the test was to execute take off, navigation through waypoints and land without remotely controlling the robot.

First experiments revealed flaws in communication system. The introduced delays made system unstable in transaction states. The UA V would loose control and crash than behaviors are switched. The necessary changes, described in section IV-B were made and test repeated.

The later test finished autonomous flight with success. The stability of behavior switching was improved at cost of switching time increasing on 50% .

C. Vision-aided control

The experiments on vision-aided control revealed limitations of use GS as a control source. We were able to track marker with hight precision, however feedback loop was running slow leading to divergent control. After certain performance modifications and a camera with a higher frame rate we were able to achieve control rate of 14.2hz. As a result we conducted an experiment with UAV hovering at set location. We observed oscillation over set point of 0.3 meters. We expect oscillation to be reduced in outdoor setting where we can take advantage of dynamic position hold functionality.

V. Future work

One of possible directions is investigate cases then UGV has uncertain path. Prediction of rendezvous location based only on initial locations of UAV and UGV will give more flexibility to a system and decrease amount of communication.

Another possible direction are cases with multiple UAVs landing on one UGV. Usage of multiple UAVs will increase scope of the system. The rendezvous location derivation and landing prioritization tasks will need to be solved.

Substantial amount of work need to be done in order to improve robustness of the system and landing behavior in particular.

VI. Conclusion

In this report the autonomous navigation and landing system is presented. Firstly, wireless communication architecture between unmanned vehicle and ground station is presented. We reveal solution capable of working with weak communication links in safe and reliable way. Second, we present coordination problem for aerial and ground robots. The goal was to derive optimal meeting location, given current positions and path of one of the robots. Third, we provide description of vision-based position estimation by means of computer vision techniques. Report provides accuracy and reliability analysis of developed algorithm. Fourth, the control architecture for vision-aided landing is presented. We describe landing and guidance using remote controller.

REFERENCES