

This item was submitted to Loughborough's Institutional Repository (<https://dspace.lboro.ac.uk/>) by the author and is made available under the following Creative Commons Licence conditions.



CC creative commons
COMMONS DEED

Attribution-NonCommercial-NoDerivs 2.5

You are free:

- to copy, distribute, display, and perform the work

Under the following conditions:

BY: **Attribution.** You must attribute the work in the manner specified by the author or licensor.

Noncommercial. You may not use this work for commercial purposes.

No Derivative Works. You may not alter, transform, or build upon this work.

- For any reuse or distribution, you must make clear to others the license terms of this work.
- Any of these conditions can be waived if you get permission from the copyright holder.

Your fair use and other rights are in no way affected by the above.

This is a human-readable summary of the [Legal Code \(the full license\)](#).

[Disclaimer](#) 

For the full text of this licence, please go to:
<http://creativecommons.org/licenses/by-nc-nd/2.5/>

Development of an Autonomous Control System for a Small Fixed Pitch Helicopter

Owen McAree, Jonathan Clarke, Wen-Hua Chen
Department of Aeronautical and Automotive Engineering,
Loughborough University,
Loughborough, United Kingdom
e-mail: {O.McAree, J.H.A.Clarke, W.Chen} @lboro.ac.uk

Abstract—The indoor test bed of the Autonomous Systems Laboratory has been used to create a control system for a small fixed pitch helicopter. This paper outlines the challenges posed by such a vehicle and the control system designed to overcome them. The control system utilized a visual tracking system to obtain state information without onboard hardware. Matlab/Simulink environment was used to facilitate rapid prototyping control development. An autopilot consisting of multi-channel PID controllers was implemented for flight tests. Failsafe strategies and gain scheduling were both investigated and implemented with the use of a graphical user interface. A number of flight tests were conducted and the results are provided. The future work of the laboratory is also covered.

Keywords—helicopter; autopilot; flight test; gain scheduling; tracking

I. INTRODUCTION

To facilitate the development of control systems and higher level algorithms for autonomous Unmanned Aerial Vehicles (UAVs) it is often necessary to move away from simulation into real world flight tests. Such tests often require customized vehicles and hardware with the associated time and cost of development. The Autonomous Systems Laboratory of the Department of Aeronautical and Automotive Engineering at Loughborough University, United Kingdom, has developed a small scale indoor test bed to utilize commercial off-the-shelf (COTS) vehicles without the requirement for custom hardware development [1]. This was done using a Vicon Visual Tracking System (VTS, [2]) which uses cameras to determine the state of the vehicle at any point in time.

The test bed uses the VTS to replace the onboard electronics traditionally used by UAVs to determine their state, requiring only the addition of reflective balls to the vehicle. The state information is made available to the Simulink control environment on a desktop PC allowing for rapid controller development. A standard radio transmitter is used to communicate the control signal to the vehicle.

Rotary UAVs have gained a lot of interest in recent years due to their greater flexibility when compared with fixed wing. Facilities such as the GTMax at the Georgia Institute of Technology [3] and the BEAR at the University of California, Berkeley [4] have demonstrated autonomous control of large outdoor model helicopters. In order for these vehicles to fly outside, onboard hardware is needed to measure attitude and position. MIT's indoor test bed, RAVEN, utilizes a similar VTS set up to avoid this onboard hardware requirement, [5]. RAVEN consists of a number of

quad-rotor vehicles, which feature simpler dynamic behavior than helicopters, allowing rapid development of a control system. The use of small scale helicopters in an indoor environment has not yet been explored.

This paper outlines the development of an autonomous control system for one such vehicle, an electric fixed pitch helicopter. The vehicle itself is first discussed, highlighting the difficulties faced by the controller. Next, the autopilot is introduced leading to the failsafe requirements. Gain scheduling is discussed with the focus on improving vehicle performance throughout a flight. Flight test results are provided and discussed. Finally, current and future work being conducted in the laboratory is discussed.

II. HELICOPTER

The type of vehicle used is a small COTS electric powered fixed pitch remote control (RC) helicopter (Fig. 1). A fixed pitch helicopter climbs by increasing the main rotor speed in contrast to collective pitch which alters the angle of attack of the blades. The fixed pitch nature of the blades results in a sluggish response to changes in height due to the large inertia of the rotor disc. In addition, fixed pitch helicopters utilize a lower tip speed than collective pitch resulting in generally lower maneuverability and greater instability. For indoor use, however, the small size of fixed pitch vehicles makes them more attractive for use. Additionally, lower blade speeds pose less of a hazard to personnel and equipment present indoors.

Control of the vehicle (either manual or autonomous) is achieved via a COTS RC transmitter which interfaces with the receiver and servo hardware fitted to vehicles as standard. The only modification required to the vehicles

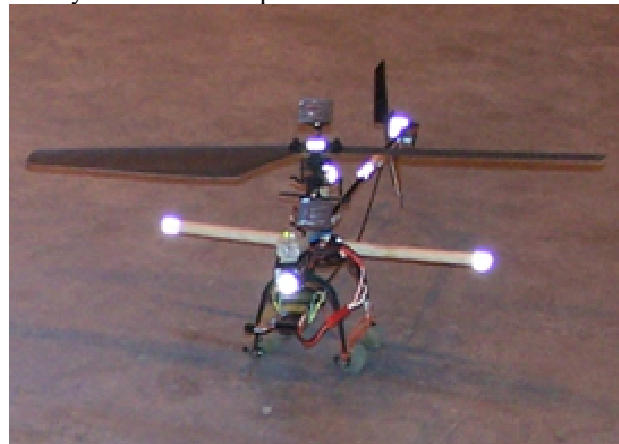


Figure 1. The 'Hummingbird' Helicopter

prior to use is the addition of a number of retro-reflectors which allow the vehicle to be tracked by the Vicon VTS. It was found that the accuracy of the VTS orientation data was proportional to the distance between outlying reflectors. For this reason a wooden bar was used to provide greater separation between lateral reflectors, seen in Fig. 1.

Manual test flying of the vehicles prior to autopilot development provided an insight into the difficulty of the control task. The equations of motion from [6] are given in (1)-(6).

$$(W/g)\ddot{u} - X_{uu}u - X_{ww}w - X_{qq}q + W\theta \cos \tau_c = X_{B_z}B_1 + (X_{\theta_0}\theta_0) \quad (1)$$

$$-Z_{uu}u + (W/g)\dot{w} - Z_{ww}w - Z_{qq}q - (W/g)V\dot{\theta} + W\theta \sin \tau_c = Z_{B_z}B_1 + (Z_{\theta_0}\theta_0) \quad (2)$$

$$-M_{uu}u - M_{ww}w - M_{\dot{w}}\dot{w} + B\ddot{\theta} - M_{qq}q = M_{B_z}B_1 + (M_{\theta_0}\theta_0) \quad (3)$$

$$(W/g)\dot{v} - Y_vv - Y_pp + (W/g)vr - Y_rr - W\phi \cos \tau_c - W\psi \sin \tau_c = Y_{A_1}A_1 + (Y_{\theta_t}\theta_t) \quad (4)$$

$$-L_vv + A\dot{p} - L_pp - E\dot{r} - L_rr = L_{A_1}A_1 + (L_{\theta_t}\theta_t) \quad (5)$$

$$-N_vv - E\dot{p} - N_pp + C\dot{r} - N_rr = N_{A_1}A_1 + (N_{\theta_t}\theta_t) \quad (6)$$

These equations illustrate the non-linear, highly coupled nature of helicopter dynamics. Equations (1)-(3) represent the longitudinal dynamics of the vehicle with u being the forward speed, w the vertical speed, q the pitch rate, θ the pitch angle, τ_c the climb angle and B_1 the longitudinal cyclic input. The θ_0 terms on the RHS of the longitudinal equations represent the collective pitch input. This term is modified for fixed pitch helicopters such that it represents the main rotor speed (which is held fixed for collective pitch). It can be appreciated, therefore, that the dynamic behavior of a fixed pitch helicopter is fundamentally no different to collective pitch. All additional terms are aerodynamic and mechanical coefficients.

Equations (4)-(6) describe the lateral dynamics with v being the lateral speed, p the roll rate, r the yaw rate, ϕ the roll angle, ψ the yaw angle, A_1 the lateral cyclic input and θ_t the tail rotor collective angle. For a fixed pitch vehicle the θ_t term is modified in a similar way to the θ_0 term to represent the fixed pitch, variable speed nature of the tail rotor.

Small helicopters also exhibit very fast natural dynamics (of the order of 0.1s) which pose a challenge for a human pilot to stabilize. Due to being very lightweight, the slightest disturbances (e.g. wind) cause large disruption to the vehicles. This, coupled with the low maneuverability, results in precise hovering being very difficult. A final observation was the close coupling of all control inputs leading to difficulties in performing accurate maneuvers.

III. AUTOPILOT

A Vicon VTS was used to provide position and orientation information at a rate of 200Hz. This information, and its derivatives, provides a full state representation of the vehicle without any approximation. The VTS uses eight infrared cameras placed around the lab, each of which detects the presence of retro-reflectors fitted to the vehicles and relays a two-dimensional image to a central processor.

This processor compares these images to build up a three dimensional representation of the lab. Provided a reflector is visible to at least two cameras the processor is able to calculate its position.

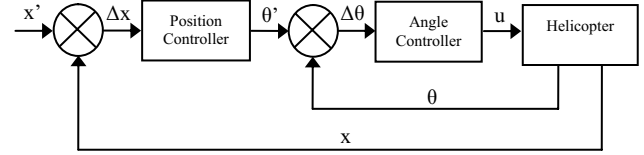


Figure 2. Pitch Controller

The processor is aware of the relative position of the reflectors on the vehicle courtesy of a model file created for each vehicle. If a sufficient number of the reflectors on the vehicle can be located the processor is able to fit the vehicle model and thus determine the position and orientation.

A custom interface was created which allows high speed data transfer between the Vicon software and the Simulink control environment, chosen to allow rapid development. The interface provides a Simulink model with the position and orientation of a vehicle at a rate of 100Hz. A nested proportional-integral-derivative (PID) control structure was used to allow intuitive tuning of control parameters to improve performance. To simplify the controller development each axis of control was considered independently.

The structure of the pitch controller is shown in Fig. 2. The first PID controller relates the error in longitudinal position to the required pitch angle. This pitch angle is compared with the measured angle and the error is transformed into a control signal via a second PID controller.

The roll controller is identical to Fig. 2 but uses lateral position and roll angles to determine a roll control signal. The height controller relates the height error to a main rotor thrust setting through a single PID controller. Yaw control is similar to height with the yaw error being related to tail rotor thrust through a single PID controller.

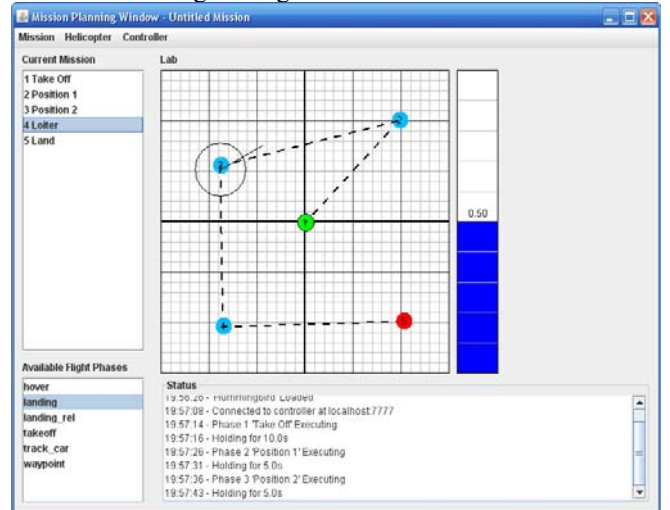


Figure 3. Graphical User Interface

Gains for each of the PID controllers were initially determined from a mathematical model of the vehicle. Due to simplifications in the model, better performance was achieved by modifying these parameters after observing the autopilot behavior during flight tests.

To communicate the control signal from Simulink to the vehicle, a USB PcTx module is used to connect the computer to a standard radio transmitter. Control of the vehicle is handed to Simulink by depressing a button on the transmitter. Releasing this button allows the vehicle to be flown manually from the transmitter.

The Simulink autopilot is capable of hovering the vehicle at a fixed location as well as translation between waypoints. Waypoints can be defined in a number of ways but it was decided to create a graphical user interface (GUI) to allow users not familiar with Simulink to operate the vehicle, Fig. 3. The GUI was written using the Java programming language due to the ease with which graphical components can be created (using the SWING libraries, [7]) and the existence of a native interface available within Matlab.

IV. FAILSAFE STRATEGIES

Introducing a GUI to allow non-specialists to operate the system poses the additional challenge of ensuring robustness in the controller. Whilst a developer is operating the system, it may be acceptable for a temporary loss of control to occur as manual recovery is possible. To ensure no such behavior when operated by a non-specialist a number of failsafe strategies were implemented.

The first of these was introduced as a result of inconsistencies in the Vicon data stream. If an insufficient number of reflectors are seen by the cameras the position and orientation data become unreliable, often defaulting to the origin. Should this condition occur, the controller is left with no knowledge of the vehicle state making a loss of control inevitable due to the natural instability of the system. To prevent this loss of control the vehicle is commanded to perform a failsafe ‘emergency’ landing if this condition is detected. The emergency landing routine immediately reduces the thrust command to zero. Due to the inertia in the rotor blades and limited operating height in the lab this results in a hard but not damaging landing. Any form of controlled landing is impossible due to the lack of feedback.

In order to remain visible to the Vicon system at all times the vehicle must be constrained to a particular working area. This requires two independent failsafe systems. The first is implemented within the GUI and prevents the user from positioning a waypoint outside of the working area. The second is implemented within Simulink which prevents the commanded position ever residing outside of the area. This second stage of failsafe is required for situations where the requested position is not being received from the GUI.

Finally, introducing a GUI which is external to the Simulink controller has benefits such as allowing execution of controller and GUI on different computers. This, however, has the downside of a potential disconnection occurring during flight. Should the GUI disconnect from the controller,

the vehicle will become stuck at its current waypoint. To mitigate this, upon disconnection the emergency landing routine implemented for the Vicon instability is triggered. This results in early termination of the flight but is preferable to the vehicle remaining stuck in a hover without a human pilot available to perform a landing.

V. GAIN SCHEDULING

To allow a user to efficiently plan a flight within the GUI a number of discrete flight phases were introduced such as take-off, hover, waypoint and landing. It was observed during development of the autopilot that the single set of controller gains used were required to compromise performance across a number of maneuvers. By separating these maneuvers at the planning phase, a set of control parameters specific to each phase can be defined. Once the vehicle has completed a phase the parameters are changed to target vehicle performance for the next phase. This gain scheduling system has been used to improve the speed of the take-off maneuver and reduce the position overshoot when maneuvering. Figure 4 shows the results including the gains used. In both cases the default gain values obtained during autopilot design were increased incrementally by 10% and a flight test performed.

To improve the take off speed the height integral gain was increased. This gain is responsible for reducing the steady state error in height. A higher value results in a faster take off but is destabilizing when the height is reached. The gain is therefore reduced back to the default value as soon as the height is reached. The upper limit for the take off gain occurred when the main rotor blades spun up too quickly, destabilizing the vehicle.

For the forward flight a large overshoot was observed with the default gains. To reduce this, the derivative gain (responsible for damping) was increased until critical damping was observed. Figure 4 also illustrates an intermediate gain value used during testing. It can be seen from this that tuning the gain values of a PID control system is a very intuitive process with an increase in derivative gain producing a corresponding decrease in overshoot. A high derivative value in hovering flight is undesirable as it imposes excessive damping on the vehicles response to disturbances making precise hovering difficult. This again demonstrates the usefulness of a gain scheduling system which can reduce the derivative gain upon completion of a maneuver such that hovering performance is improved.

In addition to improving the performance in basic maneuvers, gain scheduling allowed more aggressive maneuvers which require finer vehicle control to be introduced. These maneuvers included tracking of a moving target in addition to a fully autonomous landing on a moving platform.

VI. FLIGHT TEST RESULTS

It can be seen from the previous section that the positioning results were improved from a near 50% overshoot to virtually no overshoot (<5%) with the

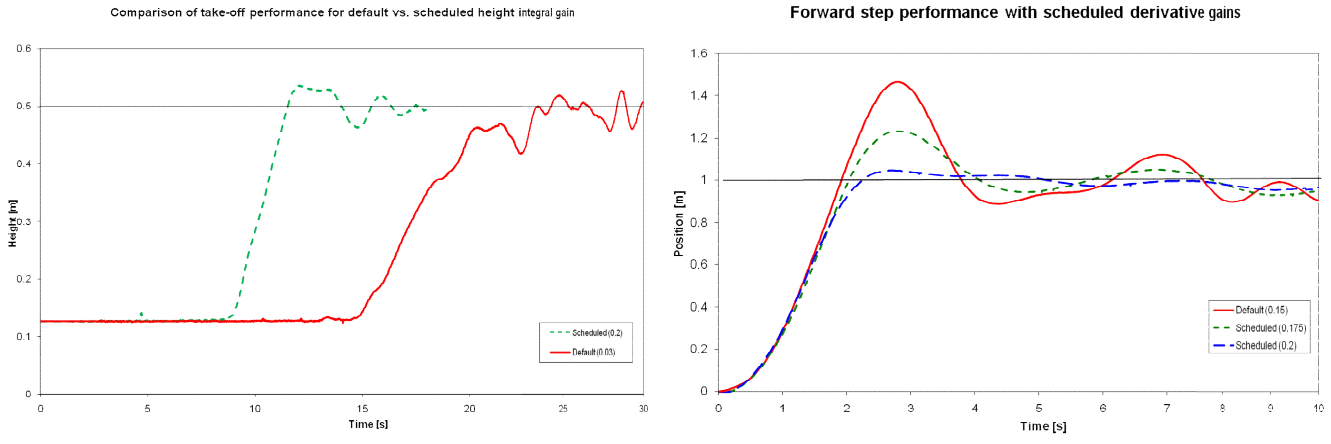


Figure 4. Gain Scheduling Performance

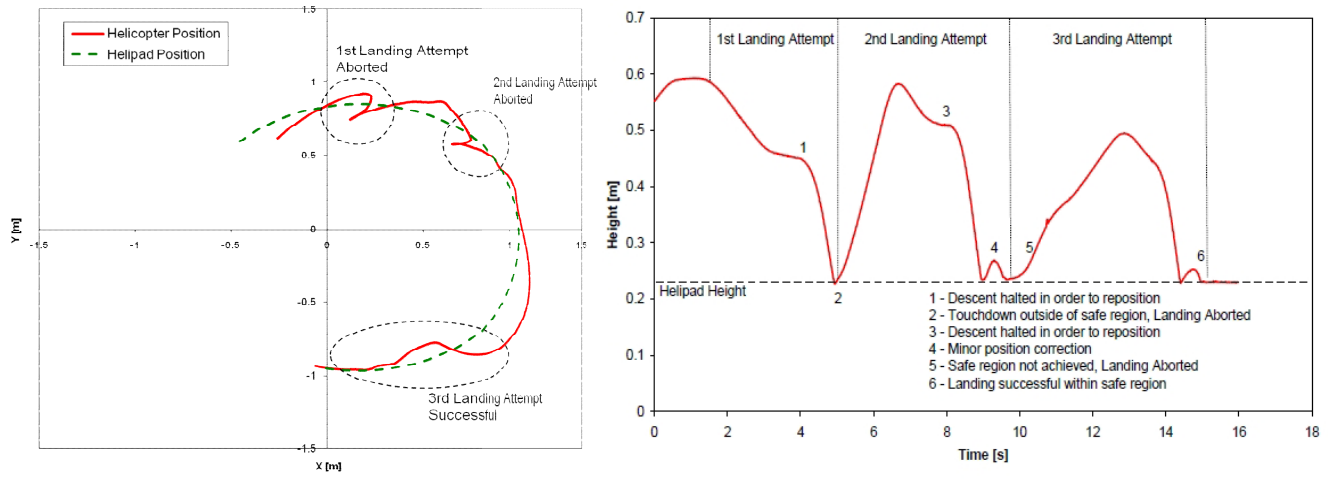


Figure 5. Landing on a Moving Platform

introduction of gain scheduling. This improvement in performance allowed the execution of complex maneuvers to be undertaken.

Figure 6 was taken during an autonomous landing of the helicopter on a mobile helipad platform. Figure 5 shows that three attempts were required to perform a successful landing. This is due to the fact that as the main rotors are slowed, in order to descend, positional control of the helicopter becomes difficult (due to the reduced maneuverability). To prevent a landing being attempted when the helicopter is poorly positioned (and possibly falling off the platform) an abort command is sent if this condition is detected. Upon receipt of the abort command the controller commands the helicopter to climb back to its holding height and position for another attempt.

The problem of reduced maneuverability during descent is compounded by the unstable nature of ground effect. For a helicopter, ground effect both increases the thrust available from the main rotors (in a similar way to the lift

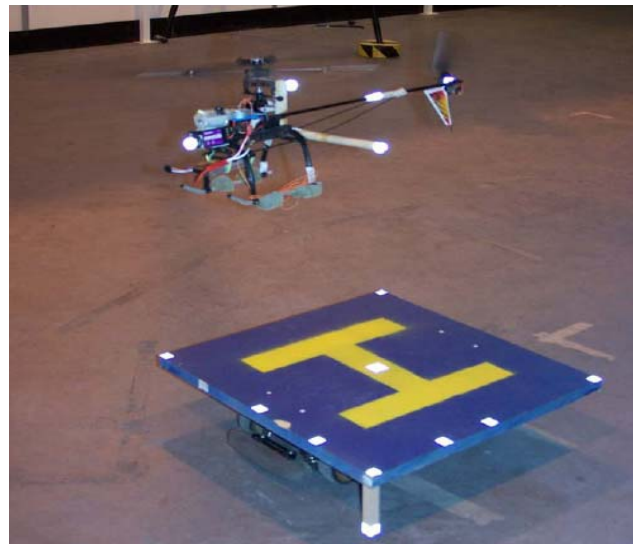


Figure 6. Helicopter and Landing Platform

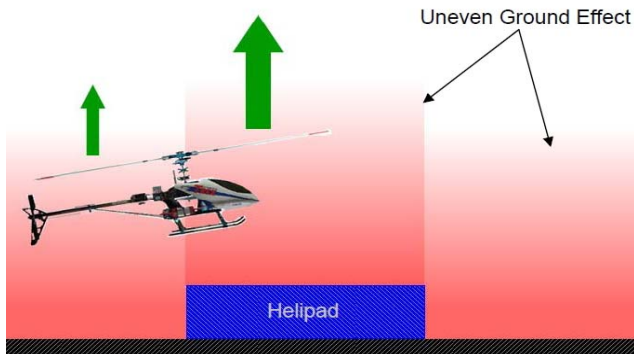


Figure 7. Illustration of the Uneven Ground Effect Problem

increase for fixed wing aircraft) but also introduces turbulence into the surrounding air. This turbulence causes the helicopter to become unstable whilst the increase in thrust requires a further reduction in rotor speed (and thus controllability) in order to descend. To further complicate matters, the presence of the solid helipad platform introduces an uneven ground effect as illustrated in Fig. 7.

Uneven ground effect provides the helicopter with a greater thrust whilst over the helipad. Any deviation from this position will result in a reduced thrust over the area of the rotor disc which is not above the helipad. The combination of these forces cause the helicopter to diverge from the helipad. When combined with the aforementioned maneuverability problems it is clear that landing on a moving platform is extremely challenging for a small fixed pitch helicopter.

VII. CONCLUSION

The results presented above represent the culmination of a large number of flight tests which have successfully demonstrated autonomous control of a small fixed pitch helicopter. The control system was developed to enable rapid deployment of small COTS vehicles without the need for customized sensor hardware. To facilitate this a VTS was used to determine the vehicle state off-board and rely this information to a computer. Simulink was used to process this data into a set of control signals. Finally a graphical user interface was created to allow non-specialist users to operate the system.

Whilst enabling rapid deployment, the use of the VTS introduced a number of issues which onboard sensing would have avoided. Firstly the ‘emergency’ condition highlighted in this paper where state information is temporarily lost and a landing must be executed. A latency problem was also experienced which caused a great deal of disruption to the progress of the project.

Despite the problems encountered with the VTS it has been successfully demonstrated that a non-specialist can command the vehicle to perform an autonomous mission, including complex maneuvers such as landing on a helipad, via the graphical interface.

VIII. FURTHER WORK

To afford better low altitude performance (as required for take-off and landing improvements) a collective pitch vehicle must be introduced. A dynamic model of a small collective pitch helicopter has just been completed and validation against hardware has begun. Once completed this model will form the basis of a control system for the vehicle. It is anticipated that the structure of the control system described in this paper can be applied directly to this new vehicle with a different set of control parameters.

Development of an onboard sensor suite is also underway. This will enable streaming of data to a ground station which can then be used to replace the VTS. Additionally, basic stabilization can be handled on board, reducing the communication burden further.

ACKNOWLEDGEMENT

The authors would like to express their thanks for financial supports for this project and the test facility from Engineering and Physical Science Research Council (EPSRC), BAE Systems and UK Higher and Further Education Council. The assistance and support from other members in Autonomous Systems Laboratory at Loughborough are also greatly appreciated.

REFERENCES

- [1] C. Liu, J. Clarke, W. H. Chen and J. Andrews, “Rapid prototyping flight test environment for autonomous unmanned aerial vehicles”, Int. J. Modelling, Identification and Control, 2010. (To be published).
- [2] “Motion Capture Systems from Vicon”, [Online]. Available: <http://www.vicon.com/> [Accessed: Oct. 26th, 2009]
- [3] E. N. Johnson, D. P. Schrage, J. V. R Prasad and G. J. Vachtsevanos, “UAV flight test programs at Georgia Tech.” Proc. AIAA “Unmanned Unlimited” Technical Conference, Workshop and Exhibit, 2004
- [4] D. H. Shim, H. Chung, H. Jin Kim and S. Sastry, “Autonomous Exploration in Unknown Urban Environments for Unmanned Aerial Vehicles” Proc. AIAA GN&C Conf., 2005
- [5] J. P. How, B. Bethke, A. Frank, D. Dale and J. Vian, “Real-time indoor autonomous vehicle test environment” IEEE Control Systems Magazine, vol. 28, 2008, pp. 51-64
- [6] A. R. S. S. Bramwell, G. Done, D. Balmford, *Bramwell’s Helicopter Dynamics*, 2nd Ed., Butterworth Heinemann, 2001, pp. 141-166
- [7] “javax.swing (Java 2 Platform SE v1.4.2)” Jun. 13th, 2003. [online]. Available: <http://java.sun.com/j2se/1.4.2/docs/api/javax/swing/package-summary.html> [Accessed: Mar. 13th, 2009]