2007 IEEE International Conference on
Robotics and Automation
Roma, Italy, 10-14 April 2007

The MIT Indoor Multi-Vehicle Flight Testbed

Mario Valenti, Brett Bethke, Daniel Dale, Adrian Frank,
James McGrew, Spencer Ahrens, Jonathan P. How, and John Vian

***Abstract*— This paper and video present the components and flight tests of an indoor, multi-vehicle testbed that was developed to study long duration UAV missions in a controlled environment. This testbed is designed to use real hardware to examine research questions related to single- and multi-vehicle health management, such as vehicle failures, refueling, and maintenance. The testbed has both aerial and ground vehicles that operate autonomously in a large, indoor flight test area and can be used to execute many different mission scenarios. The success of this testbed is largely related to our choice of vehicles, sensors, and the system's command and control architecture. The video presents flight test results from single- and multi-vehicle experiments over the past year.**

BACKGROUND

Unmanned vehicles are being used by a number of organizations to locate, observe and assess objects of interest from sophisticated operator stations miles from the area of operations. While many researchers have been discussing autonomous multi-agent operations [1], [2], more work is needed on how to perform health management for autonomous task groups. In the past, the term "health management" was used to define systems which actively monitored and managed vehicle sub-systems (e.g., flight controls, fuel management, avionics) in the event of component failures. In the context of multiple vehicle operations, this definition can be extended to autonomous multi-agent groups: teams involved in a mission serve as a "vehicle system," each vehicle is a sub-system of each multi-agent team, and so on.

As discussed in [3], many research groups have used outdoor test platforms to verify theories relating to innovative UAV concepts [4], [5], [6], [7]. In addition, there are a number of indoor testbeds that have been built to study multi-agent activities [8], [9]. These testbeds have a number of limitations that inhibit their utility for investigating the health management of UAV teams performing large-scale missions over extended periods of time. For example, outdoor platforms can only be tested during the proper weather and environmental conditions. In addition, these external UAVs also typically require a large support team, which makes long-term testing logistically difficult and extremely expensive. Furthermore, many of the indoor testbeds are either 2D or operate in 3D in a very limited flight volume.

In contrast, the MIT indoor multi-vehicle testbed is uniquely designed to study long duration missions in a controlled environment. This testbed is being used to implement and analyze the performance of techniques for embedding the fleet and vehicle health state into the mission and UAV planning. In particular, we are examining research questions related to vehicle and multi-agent health management issues, such as vehicle failures, refueling and maintenance using real hardware. The testbed is comprised of aerial and ground vehicle components, allowing researchers to conduct tests for a wide variety of mission scenarios. We have demonstrated a number of multi-vehicle coordinated test flights (using both autonomous ground and air vehicles). Currently, we are able to fly more than four air vehicles in a typical-sized room, and it takes no more than one operator to set up the platform for flight testing at any time of day for any length of time. At the heart of the testbed is a global metrology system that yields very accurate, high bandwidth position and attitude data for all vehicles in the entire room. Our testbed configuration does not require modifications to off-the-shelf radio-controlled vehicle hardware. As a result, this platform is ideal for the *rapid prototyping* of multi-vehicle mission management algorithms since it can be operated over long periods of time using one person at a fraction of the cost of what would be needed to support an external flight demonstration.

TESTBED ARCHITECTURE AND COMPONENTS

As shown in Figure 1, the testbed architecture has four major components; a mission planning level to set the system goals and monitor system progress, a task assignment level which (in general) assigns specific tasks to a vehicle or vehicle group in order to support the overall mission goals, a trajectory design level which directs each vehicle and its subsystems on how to best perform the actual tasks provided by the task processing level, and a control processing level

IEEE Member, Ph.D. candidate, Department of Electrical Engineering and Computer Science, Massachusetts Institute of Technology, Cambridge, MA 02139 valenti@mit.edu
IEEE Student Member, S.M. Candidate, Department of Aeronautics and Astronautics, Massachusetts Institute of Technology, Cambridge, MA 02139 bbethke@mit.edu
M.Eng. Candidate, Department of Electrical Engineering and Computer Science, Massachusetts Institute of Technology, Cambridge, MA 02139 ddale@mit.edu
Visiting Student, Department of Aeronautics and Astronautics, Massachusetts Institute of Technology, Cambridge, MA 02139 a_frank@mit.edu
S.M. Candidate, Department of Aeronautics and Astronautics, Massachusetts Institute of Technology, Cambridge, MA 02139 jsmcgrew@mit.edu
S.M. Candidate, Department of Mechanical Engineering, Massachusetts Institute of Technology, Cambridge, MA 02139 sahrens@mit.edu
Associate Professor of Aeronautics and Astronautics, Aerospace Controls Laboratory, Massachusetts Institute of Technology, 77 Massachusetts Ave, M.I.T. Rm 33-326, Cambridge, MA 02139 jhow@mit.edu Author to whom all correspondence should be addressed
Technical Fellow, Boeing Phantom Works, Seattle, WA 98124 john.vian@boeing.com

1-4244-0602-1/07/$20.00 ©2007 IEEE.

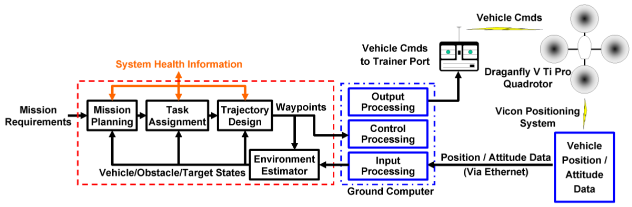

Figure 1: Testbed Architecture Block Diagram

designed to carry out the activities set by upper-levels in the system. Note that health information about each component in the system is provided to and used by each component in the architecture in the decision making process. The health monitoring components are designed to evaluate the performance of each component and provide feedback to the rest of the system (and operator) regarding its progress and mission effectiveness. This information is used by the rest of the system to potentially adjust or redirect its mission and/or task goals as the mission is taking place.

To test and demonstrate the real-time capabilities of these health management algorithms in a realistic, real-time environment, we developed a low-cost, indoor testing environment which could be used over extended periods of time in a controlled environment. The vehicles used in the platform are commercially available and off-the-shelf (COTS) R/C vehicles is designed to be durable and safe – making them suitable for an indoor flight testbed. In addition, no structural or electronics modifications were made to the air vehicles used in these flight tests, which helps to maximize flight time and reduce stress on the motor/blade components. All computing for this system is done on ground-based computers, which have two AMD 64-bit Opteron processors, 2 Gb of memory and run Gentoo Linux. The control and command processing is processed by this computer and sent over an RS-232 connection from the vehicle's control computer to the vehicle's R/C Transmitter over the transmitter's trainer port interface.

A Vicon MX camera system [10] is used to detect the vehicle's position and orientation in real-time. By attaching lightweight reflective balls to the vehicle's structure, the Vicon MX Camera system can track and compute each vehicle's position and orientation information up to 120 Hz with a 10 ms delay. This data is then transmitted via ethernet to each vehicle's ground based control computer.

In addition, the testbed is designed with an automated system task manager. Since each air vehicle in the system can take-off and land autonomously, the task manager autonomously manages every air and ground vehicle controlled in the system using these task level commands (shown in Figure 2). As a result, multi-vehicle mission scenarios (e.g., search, persistent surveillance) can be organized and implemented by the task manager autonomously. Finally, the system is designed to allow an operator to issue a command to any vehicle (at any time) through the operator interface, which includes a 3D display of the objects in the testing area and a GUI that displays vehicle health and state data, task information, and other mission-relevant data.

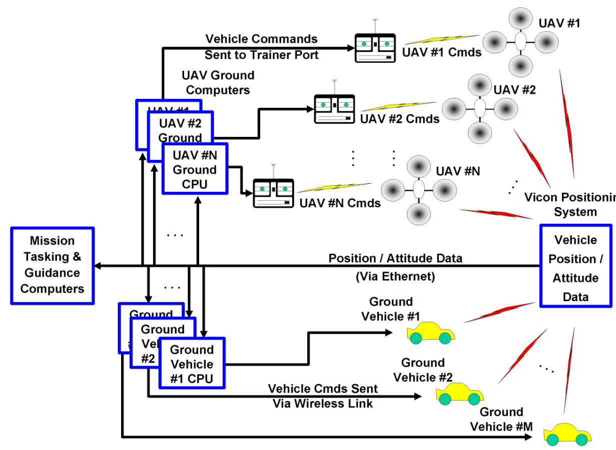

Figure 2: Testbed Command & Control Architecture

For more information and papers about this research project, visit http://vertol.mit.edu

ACKNOWLEDGEMENTS

The authors would like to thank Spencer Ahrens, Glenn Tournier and Michael Robbins for their invaluable assistance in the project. Brett Bethke would like to thank the Hertz Foundation and the American Society for Engineering Education for their support of this research. This research has been generously supported by Boeing Phantom Works in Seattle, WA. Research also funded in part by AFOSR grant FA9550-04-1-0458.

REFERENCES

[1] P. Gaudiano, B. Shargel, E. Bonabeau, and B. Clough, "Control of UAV SWARMS: What the Bugs Can Teach Us," in *Proceedings of the 2nd AIAA Unmanned Unlimited Systems, Technologies, and Operations Aerospace Conference*, San Diego, CA, September 2003.

[2] H. Paruanak, S. Brueckner, and J. Odell, "Swarming Coordination of Multiple UAV's for Collaborative Sensing," in *Proceedings of the 2nd AIAA Unmanned Unlimited Systems, Technologies, and Operations Aerospace Conference*, San Diego, CA, September 2003.

[3] M. Valenti, B. Bethke, G. Fiore, J. How, and E. Feron, "Indoor Multi-Vehicle Flight Testbed for Fault Detection, Isolation, and Recovery," in *Proceedings of the AIAA Guidance, Navigation, and Control Conference and Exhibit*, Keystone, CO, August 2006.

[4] D. Shim, H. Chung, H. J. Kim, and S. Sastry, "Autonomous Exploration in Unknown Urban Environments for Unmanned Aerial Vehicles," in *Proceedings of the AIAA Guidance, Navigation, and Control Conference and Exhibit*, San Francisco, CA, August 2005.

[5] E. King, M. Alighanbari, Y. Kuwata, and J. P. How, "Coordination and Control Experiments on a Multi-Vehicle Testbed," in *Proceedings of the IEEE American Control Conference*, 2004.

[6] D. R. Nelson, D. B. Barber, T. W. McLain, and R. W. Beard, "Vector Field Path Following for Small Unmanned Air Vehicles," in *Proceedings of the 2006 American Control Conference*, Minneapolis, MN, June 2006.

[7] G. Hoffmann, D. G. Rajnarayan, S. L. Waslander, D. Dostal, J. S. Jang, and C. Tomlin, "The Stanford Testbed of Autonomous Rotorcraft for Multi Agent Control (STARMAC)," in *In the Proceedings of the 23rd Digital Avionics Systems Conference*, Salt Lake City, UT, Nov 2004.

[8] T. J. Koo, "Vanderbilt Embedded Computing Platform for Autonomous Vehicles (VECPAV)," http://www.vuse.vanderbilt.edu/~kootj/Projects/VECPAV/, July 2006.

[9] O. Holland, J. Woods, R. D. Nardi, and A. Clark, "Beyond Swarm Intellegence: The UltraSwarm," in *Proceedings of the 2005 IEEE Swarm Intellegence Symposium*, Pasadena, CA, June 2005.

[10] Vicon, "Vicon MX Systems," http://www.vicon.com/products/viconmx.html, July 2006.