Multi-Vehicle Flight Testbed for Extended Mission Analysis, Fault Isolation, and Recovery

Mario Valenti, Jonathan P How
Massachusetts Institute of Technology
Cambridge, MA, USA 02139

John Vian
Phantom Works, The Boeing Company
Seattle, WA, USA 98124

valenti@mit.edu, jhow@mit.edu, john.vian@boeing.com

ABSTRACT

This paper presents a unique indoor multi-vehicle testbed named RAVEN (Real-time Autonomous Vehicle indoor test ENvironment) that was constructed to enable investigations of long-duration missions in a controlled environment. RAVEN provides a platform for demonstrating algorithms that embed the fleet and vehicle health state into the mission and UAV planning by enabling researchers to examine questions such as the rate and impact of vehicle failures on mission success, and what are the best strategies for performing routine refueling and maintenance using real hardware. RAVEN is comprised of both aerial and ground vehicles, allowing researchers to conduct tests for a wide variety of mission scenarios. This paper discusses this testbed infrastructure and presents flight test results from some of our most recent single- and multi-vehicle experiments. In addition, this paper presents results for an autonomously-controlled multi-vehicle persistent surveillance mission scenario.

1.0 INTRODUCTION

While many researchers have been discussing autonomous multi-agent operations [1],[2], more work is needed on how to perform multi-agent 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 [3]. Prognostic and health management techniques are being developed for new military aircraft systems to reduce future operating, maintenance, and repair costs [4]. In the context of multiple vehicle operations, we can extend this definition to autonomous multi-agent teams. In this case, teams involved in a mission serve as a "vehicle system." Then, each multi-agent team involved in the mission is a sub-system to the larger mission team. In addition, each vehicle is a sub-system of each multi-agent team, and so on.

As discussed in [5], several research groups have developed a variety of platforms to verify advanced theories and approaches for UAVs. Many of the multi-UAV platforms are built for outdoor use and examine questions related to autonomous exploration in unknown urban environments or probabilistic pursuit-evasion games [6], [7], autonomous coordination and control algorithms [8],[9], and other multi-vehicle experiments [10-12]. In

Valenti, M.; How, J.P.; Vian, J. (2007) Multi-Vehicle Flight Testbed for Extended Mission Analysis, Fault Isolation, and Recovery.
In *Platform Innovations and System Integration for Unmanned Air, Land and Sea Vehicles (AVT-SCI Joint Symposium)* (pp. 18-1 – 18-16).
Meeting Proceedings RTO-MP-AVT-146, Paper 18. Neuilly-sur-Seine, France: RTO. Available from: http://www.rto.nato.int/abstracts.asp.

Figure 1: Five Autonomous Vehicle Flight Test using RAVEN's Autonomous Mission Manager

addition, there are a number of indoor multi-vehicle platforms, most of which operate on the ground which are being used for multi-vehicle control and networking research [13]. Of the indoor platforms that have developed for flight testing, most of the vehicles have tended to be large mainly because of onboard sensing constraints, require a large area to fly and significant period of time for setup [14]. More recent indoor flying platforms have, of course, become significantly smaller. For example, researchers at Vanderbilt University built the Vanderbilt Embedded Computing Platform for Autonomous Vehicles (VECPAV) which uses two vehicles that fly indoors [15]. Also, the UltraSwarm Project at the University of Essex uses indoor aerial vehicles to examine questions related to flocking and wireless cluster computing [16].

However, as with mission-critical systems for a single vehicle, multi-agent mission planning managements systems must account for vehicle- and system-level health-related issues to ensure that these systems are cost effective to operate. The testbeds listed above have various constraints that limit their utility for investigating the health management of UAV teams performing large-scale missions over extended periods of time.

To address this problem, we have constructed a unique indoor multi-vehicle testbed named RAVEN (Real-time Autonomous Vehicle indoor test ENvironment) that is well-suited to investigate long-duration missions in a controlled environment. RAVEN is comprised of both aerial and ground vehicles, allowing researchers to conduct tests for a wide variety of mission scenarios. A unique combination of the approach for sensing the position and attitude of the vehicle combined with the choice of using commercial vehicles provides a excellent facility for demonstrating algorithms that embed the fleet and vehicle health state into the mission and UAV planning using real hardware.

Since this is an indoor testbed that uses small, unmodified electric helicopters, we have been able to fly more than five air vehicles (as shown in Figure 1) while operating multiple ground vehicles in a 8-meter by 5-meter by 3-meter flight volume, and it takes no more than one operator to set up the platform for flight testing in less than 20 minutes at any time of day for any period of time. As a result, an operator can perform a number of test flights in a short period of time, and it is routine to have a single operator command multiple UAVs during a mission.

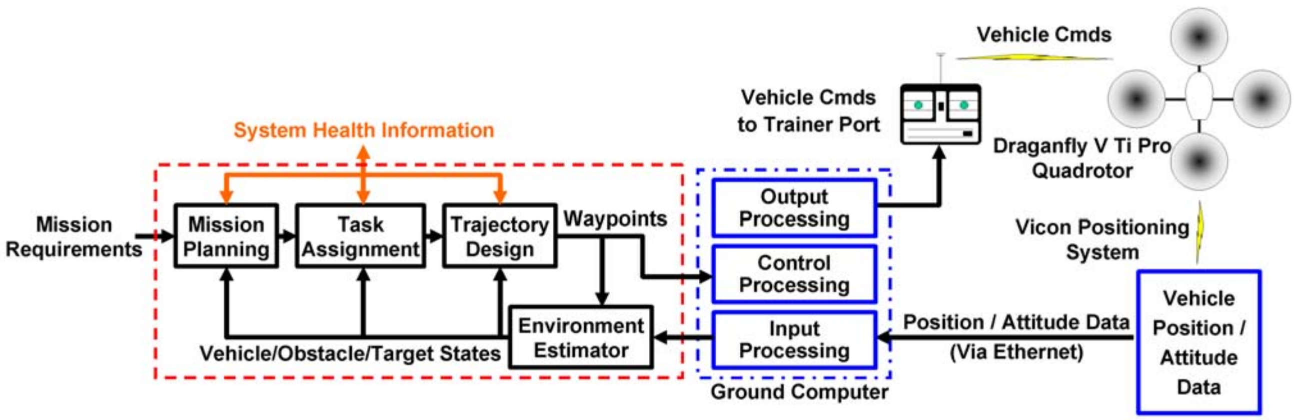


Figure 2: Integrate Mission System Architecture Block Diagram

RAVEN uses a global metrology system to acquire very accurate, high bandwidth position and attitude data for all vehicles in the entire room. Since the position markers are lightweight, the position system is able to sense vehicle position and attitude without adding any weight to the vehicles. Therefore, R/C vehicles do not require hardware modifications when used in RAVEN. In addition, we have demonstrated many types of multi-vehicle coordinated test flights, as well as extended mission scenarios (using both autonomous ground and air vehicles) to investigate very realistic mission scenarios such as persistent surveillance, convoy protection, persistent area denial, and search-and-rescue operations. Thus, 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.

2.0 PLATFORM ARCHITECTURE AND COMPONENTS

The testbed is designed to emulate the full functionality of an operational UAV system. The architecture of the system (shown in Figure 2) has the following components:

- Unmanned Aerial and Ground Vehicles
- Indoor Positioning System
- Landing / Recharge Platform
- Vehicle Ground Computers
- Mission Tasking and Navigation Computers
- Operator Interface

For flight demonstrations, the Draganflyer V Ti Pro Quadrotors [17] (as shown in Figure 1) are used for a number of reasons – they are small (~0.7m from blade tip to blade tip), lightweight (under 500g) air vehicles with a payload capacity of about 100g that can fly between 13-17 mins on one battery charge (using a 2000mAh battery) while carrying a small camera. Their four-bladed design simplifies the dynamics and

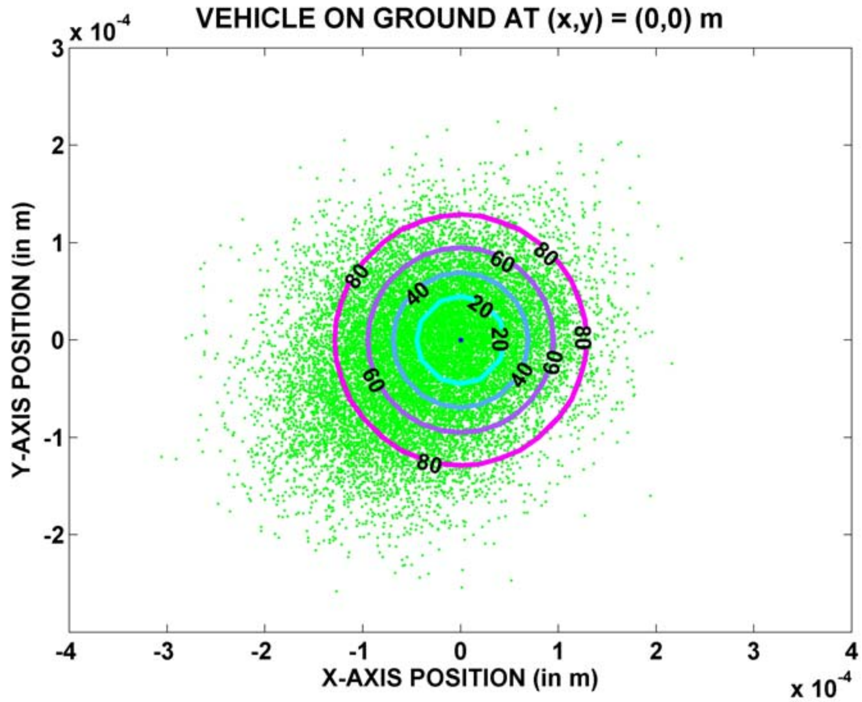


Figure 3: X-Y Scatter plot of 7 mins of measured (x, y) vehicle position at 50 Hz – rotors not turning. Note that the scale in these plots is meters x10^{-4}). The circles in this plot designate the percentage of the data points from (x,y) = (0,0) m. Note that over 80% of the points are within 0.00015 m of (x,y) = (0,0) m.

control, and the vehicle's airframe is very robust and extremely easy to repair in the event of a crash. In addition, the rotor blades are designed to fracture when they hit a solid object (thus preserving an indoor environment). Thus, these vehicles are designed to be durable and safe – making them suitable for an indoor flight testbed.

All of RAVEN's computing is done on ground-based computers, which have two AMD 64-bit Opteron processors, 2 Gb of memory and run Gentoo Linux. The control processing and command data is processed by this computer and sent over serial 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 [18] 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 and Tarsus software can track and compute the vehicle's position and attitude information up to 120 Hz with a 10 ms delay. This data is then transmitted via ethernet to the vehicle's ground based control computer.

As described above, no structural or electronics modifications were made to the Draganflyer Quadrotors used for this testing. This allows us to autonomously operate unmodified R/C vehicles without requiring any additional battery or computer payload be carried. This helps maximize the vehicle flight time and reduces stress on the motor/blade components.

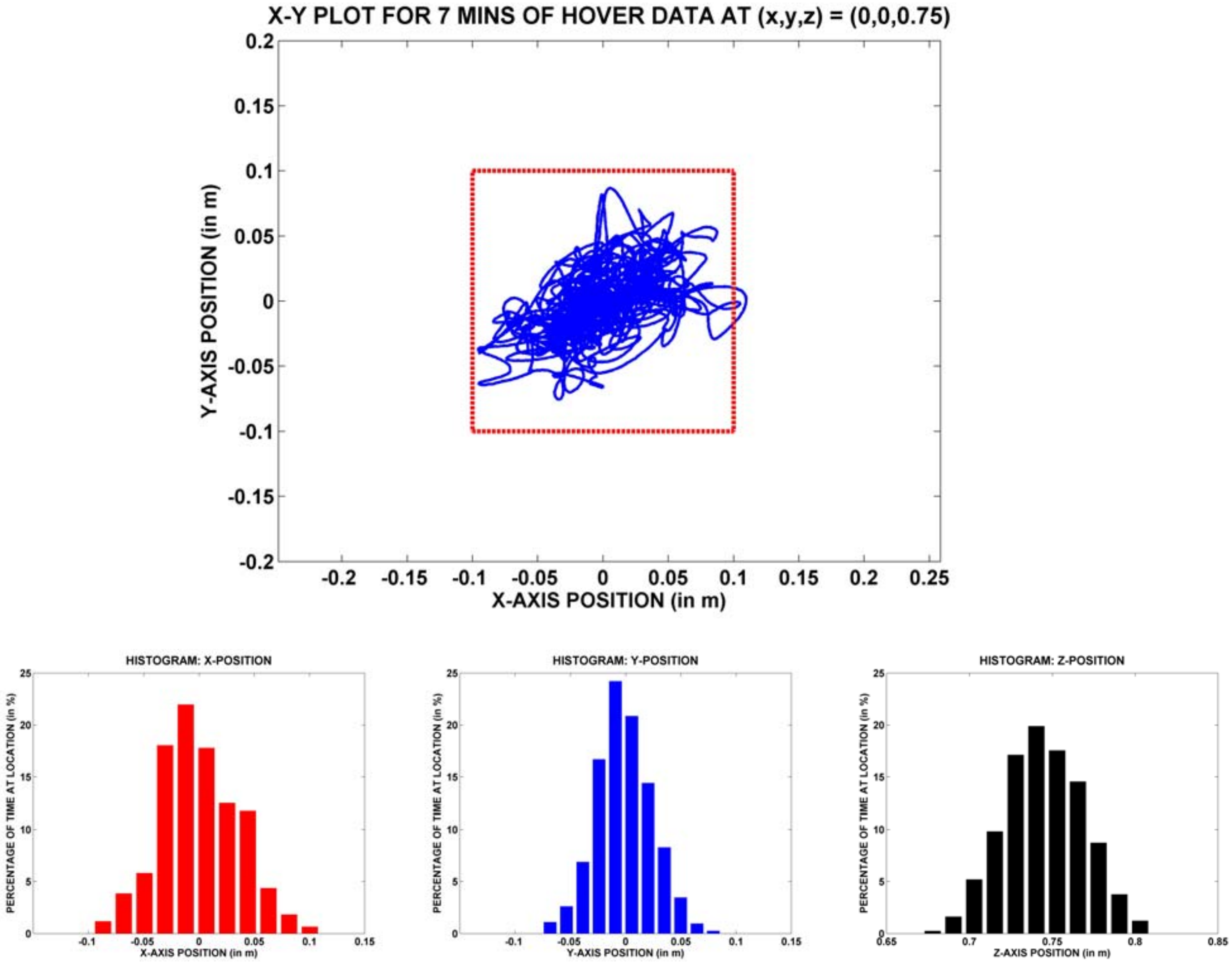


Figure 4: Single vehicle hover experiment – UAV commanded to hover at (x,y,z) = (0,0,0.75) m for 7 min: x-y plot of vehicle position (left), histograms with percentage of time at location for x-position (center) and y-position (right)

Since it is very difficult to confirm the sensor accuracy in flight, Figure 3 shows a scatter plot of the measured (x,y) position (in meters) of a quadrotor sitting on the floor at position (0,0). Note the scale on the plot – with the rotors not turning, the maximum x-position error measured by the system is 0.00032m and the maximum y-position error measured by the system is 0.00026m. Tracking multiple balls in a unique orientation on each vehicle enables the Vicon system to determine the position of the center of mass and the attitude of each air/ground vehicle that is within range, which is on the order of tens of meters.

In addition to the hardware components of this system, the integrated mission system architecture (as described above) is designed to develop and rapidly prototype hardware and software algorithms for multiple unmanned vehicle missions. Currently, our research group has developed and tested mission-level processing, task assignment and trajectory generation algorithms using RAVEN. In fact, recently, users have been developing and prototyping mission algorithms in a numerical simulation software package (Matlab) and have been able to command the vehicles in the platform using the testbed's external command interface [19]. As a

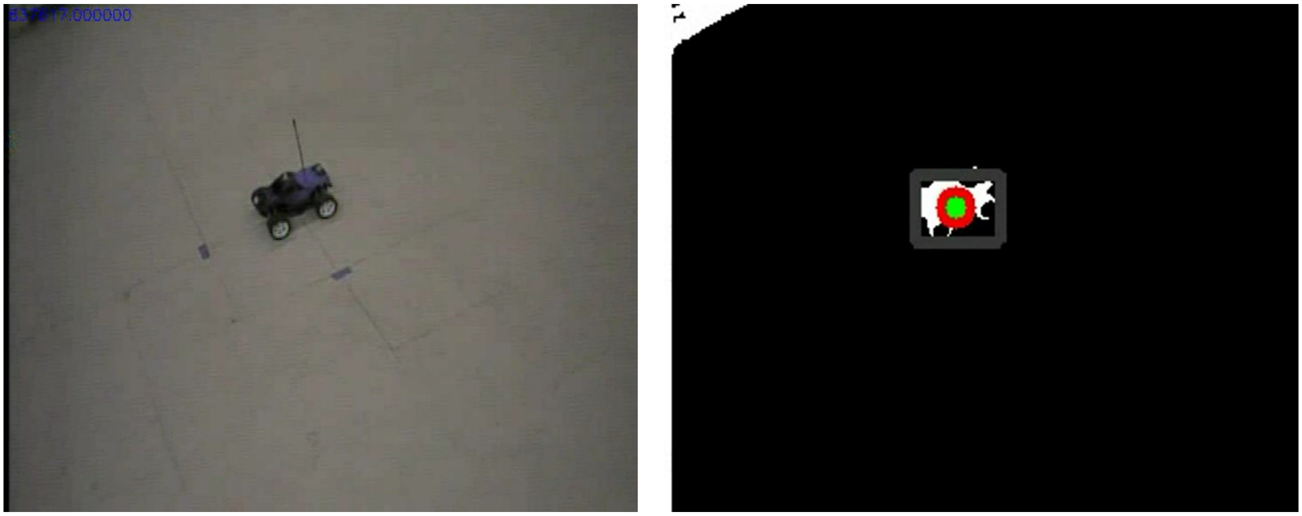


Figure 5: Vision Tracking Experiment – UAV commanded to identify ground vehicle where the raw (left) and processed (right) images from UAV [23]

result, we have successfully been able to transfer, develop, integrate, and test a variety of technologies and examine their effects in multi-vehicle operations [5],[19-24]. Thus, RAVEN provides researchers with an opportunity to develop, integrate, test and evaluate algorithms in a hardware-based at a very low cost, thus reducing the time between the development, testing and implementation of such algorithms on larger platforms.

3.0 TEST FLIGHTS AND EXPERIMENTS

Since the goal of RAVEN is to study long-duration missions in a controlled environment, the recent focus of our lab has been to ensure that RAVEN can reliably fly multiple mission sorties. As shown in Figure 1, a variety of multivehicle tests and mission scenarios have been flown using the testbed. Since January 2006, over 2000 vehicle experiments have been performed.

Typical results from a 7-min hover test are shown in Figure. In this test a single quadrotor is commanded to hold its position at (x,y,z) = (0,0,0.75) m for a 7-min period of time. Figure 4 shows three plots, including a plot of the vehicle's x- and y-positions during this test. As shown in Figure 4, the vehicle almost maintains its position inside the 20-cm dashed red box during the entire flight. The remaining plots in the figure are the histograms of the vehicle's x- and y-positions during these tests. This test clearly shows the excellent hovering performance that the quadrotors demonstrate during each mission flown..

In addition to these single-vehicle experiments, several multi-vehicle experiments and test scenarios have been conducted. These tests include, but are not limited to, formation flight tests, coordinated vehicle tests, and multi-vehicle search and track scenarios [5]. One such test is a vision-based tracking demonstration. As shown in Figure 5, a camera was mounted to a quadrotor and a small radio-controlled truck, placed at (0,0) m, was used as a target [23]. During this test flight, the vision tracking system for the air vehicle estimated the position of the ground target at the rate of the camera system (30Hz), and this data was recorded along with the raw and processed video streams (as shown in Figure 5). In the processed images, the green dot indicates an object that is being tracked and identified by the air vehicle's vision system.

In addition to quadrotors, RAVEN is being used to rapidly prototype and test other air and ground systems. For example, a foam R/C aircraft (as shown in [24] and discussed in [25]) is being used to explore the properties of an aircraft flying in a prop-hang (that is, nose-up) for the purposes of landing vertically and performing other complex maneuvers, such as perching [24]. Just as with the quadrotor, numerous hover tests were performed with the foam airplane. For example, as shown in [24] the airplane was autonomously commanded to land in a perching position on a vertical landing platform. This test was repeated twice back-to-back successfully and is discussed in detail in [25]. Other videos of these test flights are available online at http://vertol.mit.edu.

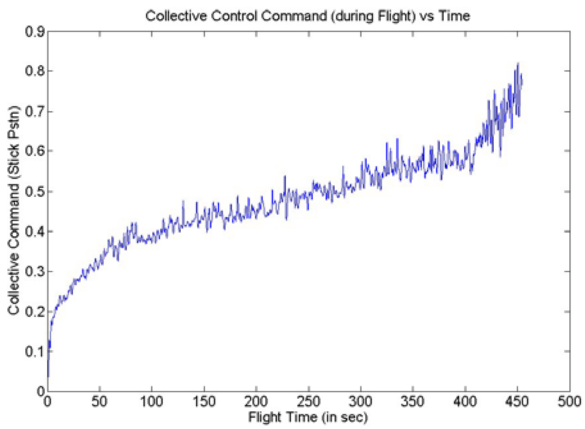
4.0 EXTENDED MISSION PLANNING

In planning and monitoring the persistent surveillance mission, the autonomous tasking system must be able to assess the status of the vehicles to determine whether a failure has occurred during the flight. In these tests, vehicle failures can result in the loss of a vehicle asset if the situation is not managed properly. In addition, most persistent surveillance mission models make the assumption that the remaining flight time for a vehicle is known or decays in a well-defined way. However, this assumption does not hold in many cases. For example, small electric-powered UAVs are powered by batteries. Since these batteries are being charged and discharged over time, they decay at different rates. This rate can change based on the type of vehicle that is used, how the battery is stressed during vehicle use, the charger used to re-charge the battery, the temperature of the environment, and many other characteristics. Therefore, one can not assume in many cases that a electric-powered vehicle with a fully-charged battery will be able to sustain a specific flight time without considering the other information provided. For this reason, knowing the current health state of the vehicle can improve the performance of the overall mission system. Therefore, in order to automate the 24-hr persistent surveillance mission, two additional components are required: 1) vehicle health monitoring – and more specifically, a battery monitor, and 2) an automatic vehicle recharging station.

4.1 Battery Health Monitoring

Most aerial vehicles have flight time limitations based on fuel and maintenance constraints. Sensors can be added to measure the vehicle's nominal health state. Using this information, health monitors can be developed to evaluate the vehicle's current capabilities. However, in some cases, *non-invasive* health monitors can also be used to evaluate a system's health without adding sensors or making changes to existing vehicle hardware. Since RAVEN uses commercially available off-the-shelf (COTS) R/C hardware, sensors can not be added without making intrusive modifications to the vehicle hardware. For this reason, a portion of our research effort is focused on developing and testing non-invasive health monitors with the COTS hardware used in our testbed.

For example, one of the main issues in using electric powered vehicles is that an electric-powered vehicle's endurance scales with motor power consumption and battery size, and thus battery weight (which has a negative effect on vehicle endurance). Although battery technology has improved in recent years (for example, a 3-cell, 11.1V 1320mAh Li-Poly battery from Thunder Power Batteries [26] is approximately 100g), an electric-powered air vehicle's flight time is largely impacted by the vehicle's lift capabilities. In addition, the average flight time of electric-powered helicopters and quadrotors are limited by the types of motors used and desired payload capacity.



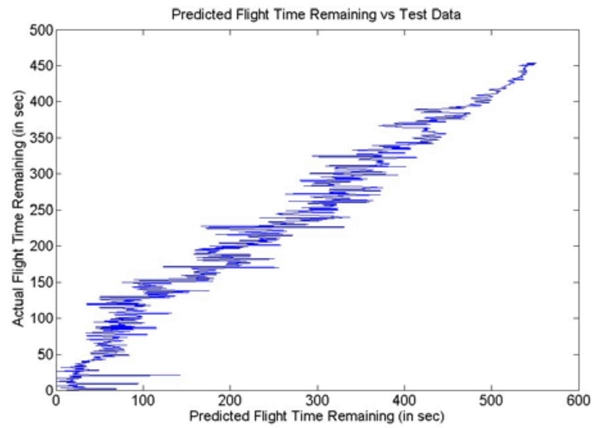


Figure 6: Collective Control Command during an Autonomous Quadrotor Hover Test (left), Predicted vs Actual Remaining Flight Time for a Hover Test using Non-invasive Battery Monitor

Past research into battery monitoring and state estimation has focused on using direct, invasive measurements of current flow and voltage level to calculate a battery's state of charge (SOC). Most of this research focuses on calculating SOC using complex analytical models of internal battery dynamics [27],[28]. However, these approaches require significant knowledge of battery properties and internal dynamics. Some recent research has sought to simplify the construction of battery SOC models by using machine learning techniques using voltage and current measurements from the battery [29]. A learning approach can be advantageous because it does not require knowledge of internal battery chemistry and can be easily extended to multiple battery chemistries.

As part of our on-going research, the relationships between a vehicle's flight capabilities, power system health, propeller wear, battery charge and other parameters are being examined using electric-powered quadrotor vehicles [5]. Flight testing has demonstrated that many of these parameters can be evaluated while observing the vehicle in a simple hover. For example, that data from these flight tests has shown there is a strong correlation between the battery voltage and the average collective stick position value. As the voltage of the battery decreases over time (due to battery use). Figure 7 shows that as the battery's charge level decreases, the average collective command must increase over time for the vehicle to maintain its current position. Notice that the collective command increases rapidly initially (during take-off) and steadily increases almost linearly until the vehicle's battery loses charge rapidly near the end of the flight.

To generate an estimate of the vehicle's remaining flight time non-invasively using the vehicle's altitude and collective input position, a support vector regression (SVR) model was generated using the vehicle's current altitude and collective position from a series of test flights to estimate the vehicle's remaining flight time. The main idea behind this approach is that a model can be created using experimental data from input-output pairs. These pairs are used to generate predict the output to a system given an input without explicitly knowing the model of the system [30-32]. Note that this model can be improved by regenerating the SVR model based on new data, thus ensuring that the model is up-to-date and representative of the current situation.

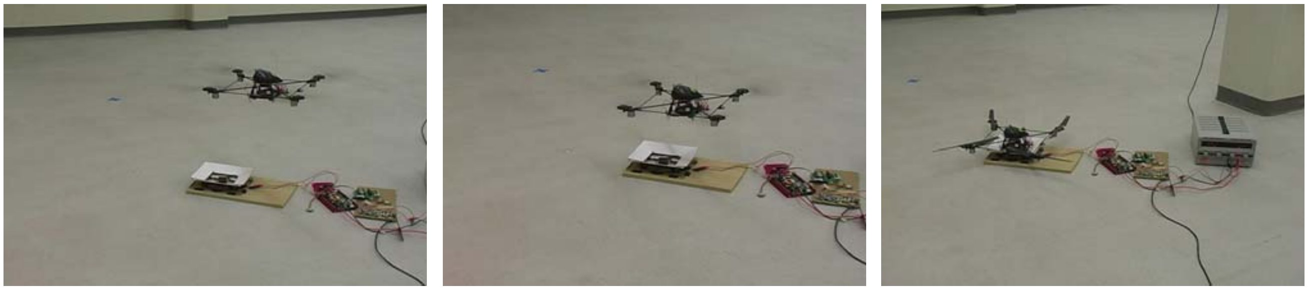


Figure 7: Fully Autonomous Landing and Recharge Demonstration

Over 10,000 data points were used to generate the SVR battery model. This battery model was tested against actual vehicle hover flights as shown in Figure 7. Here, the predicted flight time generated from the model provides a reasonable estimate of the vehicle's remaining flight time. In fact, by filtering the vehicle's collective stick position, a better estimate of the vehicle's remaining flight time is generated. This model is currently being used to command the quad rotor vehicles to automatically land and recharge the quadrotor vehicles during extended flight activities.

4.2 Battery Charging Station

Even as electric-powered autonomous vehicles and their support systems become smarter, they are fundamentally limited by the capacity of onboard batteries that power the vehicles. As a result, autonomous health management hardware and software allow vehicles to determine the battery's current status and decided when a vehicle must land to replace or recharge itself before continuing its mission. Ground platforms have been developed to allow robots to recharge during operations. For example, at the University of Tsukuba, researchers constructed an autonomous ground vehicle and recharge system in order to facilitate autonomous ground vehicle navigation and control experiments [33]. The system was tested by running an autonomous vehicle non-stop for one week. During the week-long experiment, over one thousand recharge dockings were successfully accomplished. However, while docking and recharge stations are now commonplace for ground vehicles, as of the writing of this paper, the authors have not found an instance of an autonomous aerial docking and recharge in the literature before reporting it in [5].

Therefore, to conduct research into autonomous, multi-vehicle, persistent surveillance mission applications, an integrated autonomous recharge landing platform and system was designed. As shown in Figure 8, this landing platform system has autonomously taken off, landed on the recharge platform, recharged the battery, and taken off fully autonomously. Pictures of the first landing attempt performed by the system at the end of July 2006, and recent pictures of an automated landing and recharge test are shown in Figure 8. In this test, the landing pad was placed at (0,5,0)m in the room and the vehicle was commanded to land after hovering for several minutes. In this landing demonstration, the vehicle was commanded to land after hovering above the landing location for five seconds. This waiting time is designed to allow transients in the x- and y-position controller to settle out before continuing with the landing decent. Notice that once the vehicle begins its decent, the vehicle corrects its position as necessary to ensure that it can land on the recharging location.

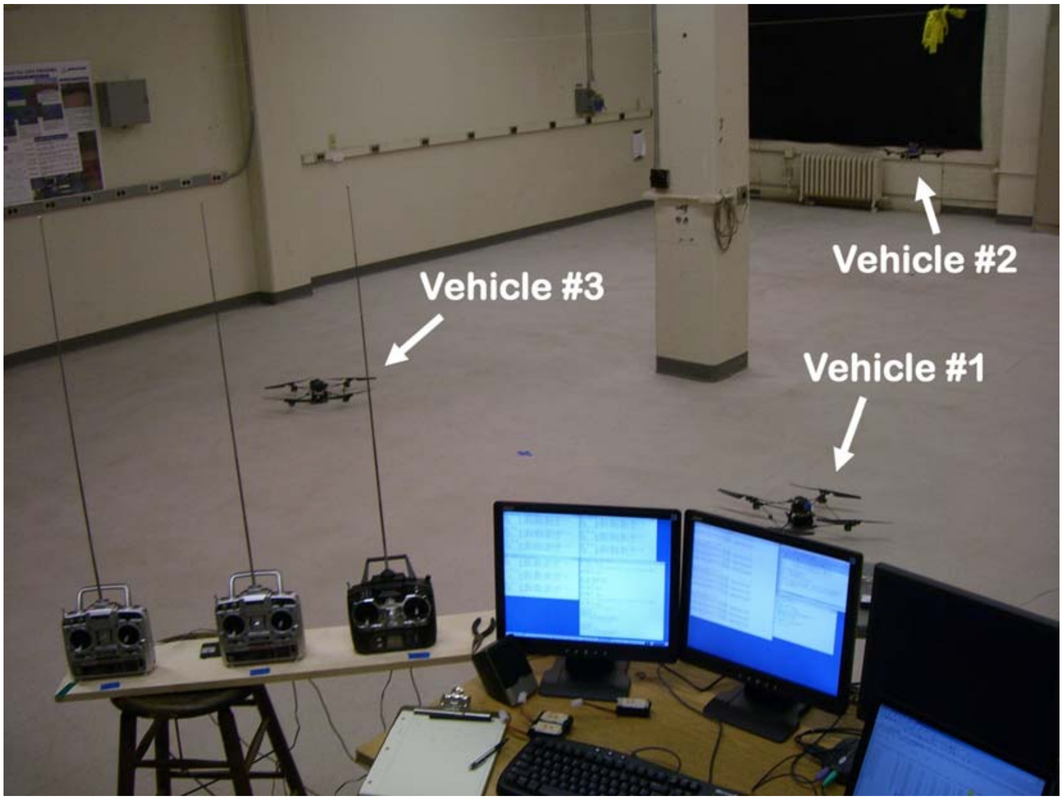


Figure 8: Automated 1.5 Hr Persistent Surveillance Mission (PSM) with Three Autonomous Vehicles

4.3 Results

Using the battery monitors described above, a 1.5 hour persistent surveillance mission with three vehicles was setup in the laboratory as shown in Figure 9. In this test, the tasking system was responsible for tasking the vehicles to take-off, fly to the surveillance location, and return to base when their batteries were depleted. The tasking system was also responsible for directing the operator to change the batteries of vehicles that had just landed. An operator acknowledgement was added as a safety feature to ensure that the tasking system would not take-off any vehicle in the battery maintenance area while the operator was in the space.

Data from the 1.5 hour persistent surveillance mission with three vehicles is shown in Figure 10. During the 93 min test, there were two vehicle failures, both occurring with Vehicle #1. Approximately 45 mins into the test, the tasking system commanded Vehicle #1 to take-off, however the vehicle was unresponsive. As a result, since the tasking system did not observe a response from Vehicle #1, it then commanded Vehicle #2 to take Vehicle #3's place. This delay resulted in a 25 sec gap in surveillance coverage. In addition, after the next round of rotations, approximately 73 mins into the test, Vehicle #1's low-battery alarm went off as Vehicle #2 approached the surveillance location, resulting in a 3 sec gap in coverage. For the duration of the test, coverage was maintained over the surveillance location for over 99.5% of the 93 min flight demonstration.

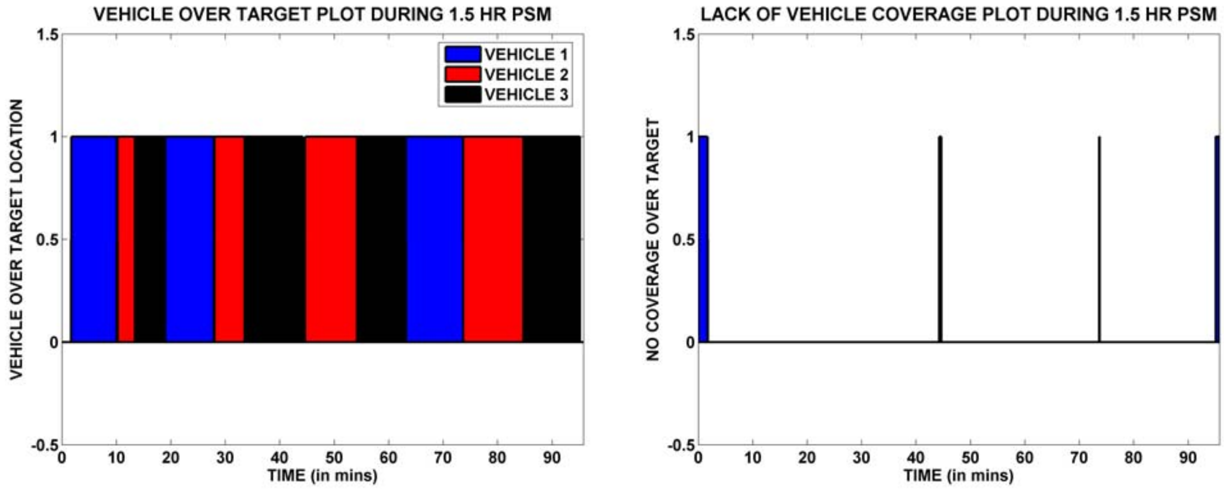


Figure 9: Coverage plot showing the vehicles flying the mission (left), Lack of Coverage Time History during Test (right)

Using the battery monitor and recharge system described above, a fully-autonomous 24 hour test was performed with a single vehicle in late March 2007. In this test, a single quadrotor vehicle flew above its recharge platform between 10 and 13 mins before landing to recharge its batteries. The quadrotor used in this test uses brushless motors, thereby making it more energy efficient than vehicles using conventional electric motors.

This flight-recharge sequence occurred 21 times during the 24 hr period. This test was performed without any operator interaction and marks the first time in the literature where an air vehicle was able to perform a routine maintenance activity (such as refuelling) multiple times over a 24 hr period. As shown in Figure 10, the red line represents the recorded flight time of the vehicle and the blue line represents the recorded recharge time of the vehicle during each flight-recharge cycle. Note that there is a 5 min wait period between the time the vehicle lands and when the recharge sequence begins. This period of time allows the battery to cool down before charging.

Currently, improvements in the battery monitor are being introduced to enable a fully-autonomous 24-hr persistent surveillance test (with automatic recharge) using multiple vehicles.

5.0 CONCLUSION

RAVEN offers government, commercial and academic organizations a low-cost flight test platform for the rapid prototyping of multi-vehicle mission algorithms and vehicle hardware. Since RAVEN is very robust, users can execute multiple missions in a short period of time with minimal setup and organization between tests. Thus, this platform will be a very attractive alternative to the existing testing methods because multi-vehicle tests can be performed using this real-time platform at a fraction of the cost. Videos of past vehicle tests are available online at http://vertol.mit.edu.

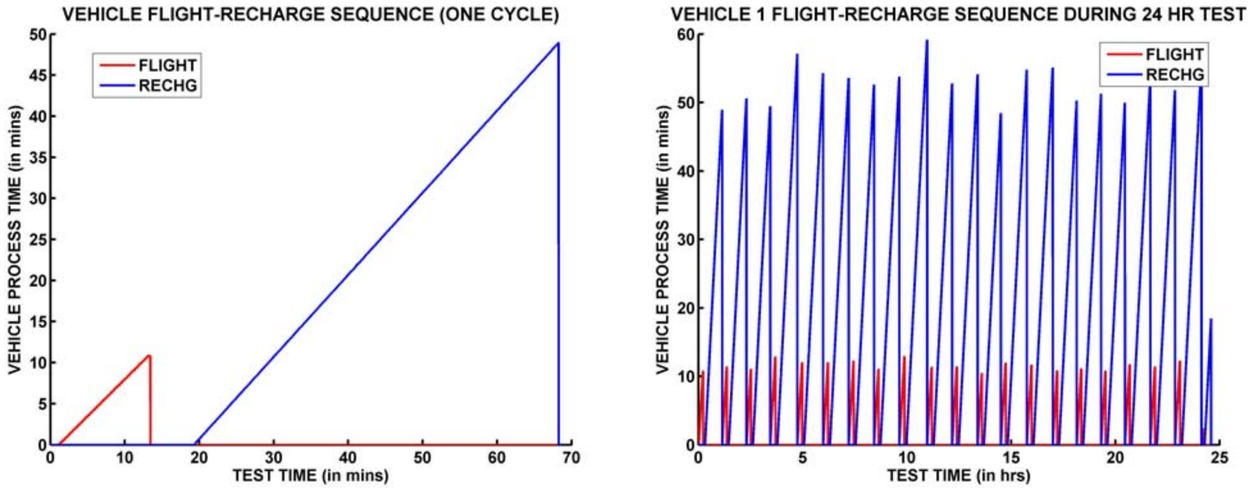


Figure 10: Flight-Recharge sequence plots from a fully-autonomous 24 hr test

6.0 ACKNOWLEDGEMENTS

The authors would like to thank Brett Bethke, Daniel Dale, Adrian Frank, James McGrew, and Spencer Ahrens for their assistance in the project. This research has been generously supported by the Boeing Company under the guidance of Dr. John Vian at the Boeing Phantom Works, Seattle, WA. Research also funded in part by AFOSR grant FA9550-04-1-0458.

7.0 BIBLIOGRAPHY

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

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

[3] Fudge, M., Stagliano, T., & Tsiao, S., *Non-Traditional Flight Safety Systems and Integrated Vehicle Health Management Systems*, Produced for the Federal Aviation Administration, ITT Industries, Advanced Engineering and Sciences Division, 2560 Alexandria Drive, Alexandria, VA 22303, August 2003.

[4] Becker, K.C., Byington, C.S., Forbes, N.A., & Nickerson, G.W., *Predicting and Preventing Machine Failures*, The Industrial Physicist, Vol. 4, No. 4, December 1998, pp. 20–23.

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

[6] Shim, D., Chung, H., Kim, H.J., & Sastry, S., *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.

[7] Vidal, R., Shakernia, O., Kim, H.J., Shim, H., & Sastry, S., *Multi-Agent Probabilistic Pursuit Evasion Games with Unmanned Ground and Aerial Vehicles*. IEEE Transactions on Robotics and Automation, Vol. 18, No. 5, 2002, pp. 662–669.

[8] How, J. P., King, E., & Kuwata, Y., *Flight Demonstrations of Cooperative Control for UAV Teams*, AIAA 3rd Unmanned Unlimited Technical Conference, Workshop and Exhibit, Chicago, IL, September 2004.

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

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

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

[12] Hoffmann, G., Waslander, S.L., & Tomlin, C., *Distributed Cooperative Search using Information-Theoretic Costs for Particle Filters, with Quadrotor Applications*, In the Proceedings of the AIAA Guidance, Navigation, and Control Conference and Exhibit, Keystone, CO, August 2006.

[13] Vladimerouy, V., Stubbs, A., Rubel, J., Fulford, A., Strick, J., & Dullerud, G., *A Hovercraft Testbed for Decentralized and Cooperative Control*, Proceedings of the 2004 American Control Conference, Boston, MA, July 2004.

[14] Olsen, E.A., Park, C-W., & How, J.P., *3D Formation Flight Using Differential Carrier-Phase GPS Sensors*, ION Navigation, Vol. 46, No. 1, 1999.

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

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

[17] Draganfly Innovations Inc., *Draganfly V Ti Pro Website*, Available online at http://www.rctoys.com/draganflyer5tipro.php, January 2006.

[18] Vicon, *Vicon MX Systems*, Available online at http://www.vicon.com/products/viconmx.html, July 2006.

[19] Kuwata, Y., *Trajectory Planning for Unmanned Vehicles using Robust Receding Horizon Control*, Ph.D. Thesis, Massachusetts Institute of Technology, February 2007.

[20] Culligan, K., Valenti, M., Kuwata, Y., & How, J.P., *Three-dimensional flight experiments using on-line mixed-integer linear programming trajectory optimization*, To appear in the Proceedings of the 2007 American Control Conference, New York, NY, June 2007.

[21] Valenti, M., Bethke, B., How, J.P., & Vian, J., *Embedding Health Management into Mission Tasking for UAV Teams*, To appear in the Proceedings of the 2007 American Control Conference, New York, NY, June 2007.

[22] Tournier, G., Valenti, M., How, J.P., & Feron, E., *Estimation and Control of a Quadrotor Vehicle using Monocular Vision and Moire Patterns*, Proceedings of the AIAA Guidance, Navigation, and Control Conference and Exhibit, Keystone, CO, August 2006.

[23] Bethke, B., Valenti, M., How, J.P., & Vian, J., *Cooperative Vision Based Estimation and Tracking Using Multiple UAVs*, Proceedings of the Conference on Cooperative Control and Optimization, Gainsville, FL, January 2007.

[24] Valenti, M., Bethke, B., Dale, D., Frank, A., McGrew, J., Ahrens, S., How, J. P., Vian, J., *The MIT Indoor Multi-Vehicle Flight Testbed*, Proceedings of the 2007 IEEE International Conference on Robotics and Automation, 10-14 April, 2007, Rome, Italy.

[25] Frank, A., Valenti, M., Levine, D., & How, J.P., *Hover, Transition, and Level Flight Control Design for a Single-Propeller Indoor Airplane*, To appear in the Proceedings of the AIAA Guidance, Navigation, and Control Conference and Exhibit, Myrtle Beach, SC, August 2007.

[26] Thunder Power Batteries, *Thunder Power Batteries Website*, Available oneline at http://www.thunderpowerbatteries.com/, September 2006.

[27] Plett, G. L., *Extended Kalman filtering for battery management systems of LiPB-based HEV battery packs*, Journal of Power Sources, Vol. 134, 2004, pp. 277–292.

[28] Rong, P. & Pedram, M., *An Analytical Model for Predicting the Remaining Battery Capacity of Lithium-Ion Batteries*, IEEE Transactions on VLSI Systems, Vol. 14, No. 5, 2006, pp. 441–451.

[29] Hansen, T. & Wang, C.-J., *Support vector based battery state of charge estimator*, Journal of Power Sources, Vol. 141, 2005, pp. 351–358.

[30] Vapnik, V., *Statistical Learning Methods*, J. W. Wiley and Sons, 1998.

[31] Gunn, S. R., *Support Vector Machines for Classification and Regression*, Technical Report, University of Southhampton, May 1998.

[32] Smola, A. J. & Schölkopf, B., *A Tutorial on Support Vector Regression*, Produced as part of the ESPRIT Working Group in Neural and Computational Learning II, NeuroCOLT2, October 1998.

[33] Hada, Y. & Yuta, S., *A First-Stage Experiment of Long Term Activity of Autonomous Mobile Robot Result of Respective Base-Docking Over a Week*, Lecture Notes in Control and Information Sciences: Experimental Robotics VII , Vol. 271, 2001, pp. 229–238.

[34] Gurdan, D., Stumpf, J. C., Achtelik, M., Doth, K.-M., Hirzinger, G., Daniela, R., *Energy-Efficient Autonomous Four-Rotor flying Robot Controlled at 1 Khz*, Proceedings of the 2007 IEEE International Conference on Robotics and Automation, 10-14 April, 2007, Rome, Italy.