

# Design, Development, and Testing of a UAV Hardware-in-the-Loop testbed for Aviation and Airspace Prognostics Research

Chetan Kulkarni\* and Chris Teubert<sup>†</sup> and George Gorospe<sup>‡</sup>  
*SGT, Inc. NASA Ames Research Center, Moffett Field, CA 94035, USA*

Drew Burgett<sup>§</sup>  
*Christopher Newport University, Newport News, VA*

Cuong C. Quach<sup>¶</sup>  
*NASA Langley Research Center, Hampton, Virginia 23681, USA*

Edward Hogge<sup>||</sup>  
*Northrop Grumman Technical Services, NASA Langley Research Center, Hampton, Virginia 23681, USA*

## Abstract

THE airspace is becoming more and more complicated, and will continue to do so in the future with the integration of Unmanned Aerial Vehicles (UAVs), autonomy, spacecraft, other forms of aviation technology into the airspace. The new technology and complexity increases the importance and difficulty of safety assurance. Additionally, testing new technologies on complex aviation systems & systems of systems can be very difficult, expensive, and sometimes unsafe in real life scenarios. Prognostic methodology provides an estimate of the health and risks of a component, vehicle, or airspace and knowledge of how that will change over time. That measure is especially useful in safety determination, mission planning, and maintenance scheduling. The developed testbed will be used to validate prediction algorithms for the real-time safety monitoring of the National Airspace System (NAS) and the prediction of unsafe events. The framework injects flight related anomalies related to ground systems, routing, airport congestion, etc. to test and verify algorithms for NAS safety. In our research work, we develop a live, distributed, hardware-in-the-loop testbed for aviation and airspace prognostics along with exploring further research possibilities to verify and validate future algorithms for NAS safety.

The testbed integrates virtual aircraft using the X-Plane simulator and X-PlaneConnect toolbox, UAVs using onboard sensors and cellular communications, and hardware in the loop components. In addition, the testbed includes an additional research framework to support and simplify future research activities. It enables safe, accurate, and inexpensive experimentation and research into airspace and vehicle prognosis that would not have been possible otherwise. This paper describes the design, development, and testing of this system. Software reliability, safety and latency are some of the critical design considerations in development of the testbed. Integration of HITL elements in the development phases and verification/ validation are key elements to this effort.

---

\*Intelligent Systems Division, Discovery and Systems Health Area, MS 269-3, AIAA Member.

<sup>†</sup>Intelligent Systems Division, Discovery and Systems Health Area, MS 269-3, AIAA Member

<sup>‡</sup>Intelligent Systems Division, Discovery and Systems Health Area, MS 269/3, AIAA Member

<sup>§</sup>Safety Critical Avionics Systems Branch, Graduate Student

<sup>¶</sup>Safety Critical Avionics Systems Branch

<sup>||</sup>Safety Critical Avionics Systems Branch, AIAA Member

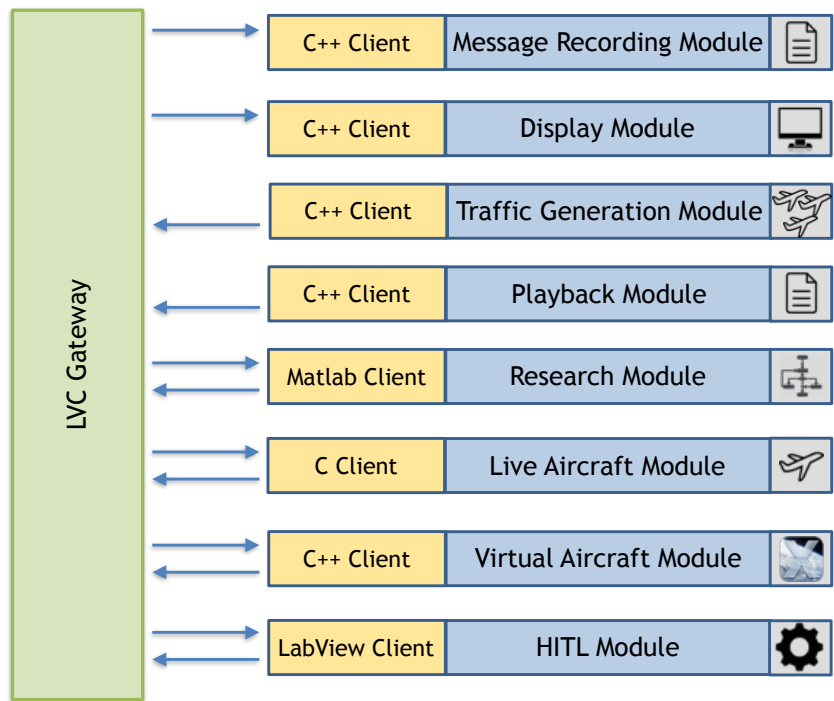


Figure 1. Schematic of the Virtual Lab Testbed

## I. Introduction

The goal of this work is to run complex experiments involving real and virtual aircraft by connecting different airborne assets, virtual aircraft simulations along with hardware-in-the-loop elements. The integration of each of these elements is through a cloud based service connecting the Facilities/Labs around the country providing unique tools and skill sets.

The virtual laboratory testbed infrastructure is developed to be scalable, allowing for plug-and-play addition of field and lab assets from new operators. This unique fusion of virtual aircraft, UAVs, hardware-in-the-loop elements, and prognostic research tools opens up many opportunities for future research into providing a safer and more efficient intelligent airspace for the future.

The testbed adds unique capabilities of incorporating hardware-in-the-loop components with real and virtual aircraft and providing a framework for research on the impact and interaction of various assets in the airspace and the future of air traffic control. Integrating hardware-in-the-loop components allow for unique studies of how the behavior of aircraft components plays into complex local airspace interactions as shown in the schematic of Fig.1. This framework is designed to conduct research for NAS safety by minimizing the amount of work required to integrate complex live and simulated elements in a single operating environment use this powerful tool.

The research module is designed to seamlessly and effectively integrate closed-loop local airspace safety, control and prognostic decision making algorithms into the experiment. The framework includes tools for integration, development, research, benchmarking, and testing for rapid feasibility studies of innovative airspace concepts. The framework will be primarily designed as a cloud-resource with a web interface, taking advantage of the increased access, and elimination of the need for each center to have its own server, eliminating duplication of resources.

The paper is organized as follows. Section II mentions the previous research work and current state of the art work. Sections III and IV present the testbed Architecture framework and testbed modules, respectively. Section V demonstrates the implementation of this framework in simulation through an example and results observed. Finally, Section VI concludes the paper and presents future work.

## II. Background and Current Research

There is an increasing desire for a capability to fly UAVs in the National Airspace System (NAS). This represents a new capability that will provide a variety of applications services in both the public and commercial aviation sector. There is a lack of common understanding for the requirements to safely operate UAVs in the NAS and with increasing demand to fly them, it will be very critical to implement safety rules and regulations.

NASA is conducting research in the areas of Sense and Avoid/Separation Assurance Interoperability (SAA), Human Systems Integration, Communication, and Certification to integrate UAVs in the NAS. To accomplish this task, LVC - Live Virtual Constructive Distributed Environment (LVC-DE) project was implemented to conduct a series of Integrated Human-in-the-Loop (IHITL) and Flight Test activities which integrate key concepts, technologies and/or procedures in a relevant air traffic environment.<sup>1</sup> This is an ongoing project work to improve on the NAS safety where several integrated events build on the technical achievements, fidelity and complexity of the previous tests and technical simulations. The studies and results from the LVC project has resulted in a body of evidence that supports the development of regulations governing the access of UAVs into the NAS.<sup>2,1</sup> A central component of this, the LVC Gateway, was utilized for this project.

In addition to the LVC tool we integrated the NASA Airborne Science Mission Tools Suite (MTS)<sup>a</sup> which is a collection of web-based tools to assist with the planning, operations, and overall management of airborne missions. The main objectives of the MTS are (a) to support tactical decision-making and distributed team situational awareness during a flight; (b) to facilitate team communication and collaboration throughout the mission life-cycle; and (c) to both consume and produce visualization products that can be viewed in conjunction with the real-time position of aircraft and airborne instrument status data. The intent of the system is to encourage more responsive and collaborative measurements between instruments on multiple aircraft, satellites, and on the surface in order to increase the scientific value of the measurements, and improve the efficiency and effectiveness of flight missions. MTS is a product of NASA's Airborne Science Program. The MTS contains a core set of tools with capabilities such as: remote monitoring real-time aircraft location, viewing of current and archived flight tracks, ability to add information overlays from a curated product registry, team communication and collaboration tools etc.

In our earlier work<sup>3,4</sup> we used an Edge 540 as seen in Fig.2, which is a scaled aircraft used to verify and validate testing procedures and prognostic algorithms that are intended to build trust in predictions of remaining flying time prior to actual flight testing. The philosophy behind the testing procedure is to translate system performance and safety goals into requirements for an alarm that warns system operators when the estimated remaining flying time falls below a certain threshold. Ground testing of the actual vehicle provides the closest possible testing conditions short of actual flight and captures some of the variation that the powertrain hardware and that the pilot may introduce while avoiding the risks inherent in flight. For instance, the batteries may be drained to a lower capacity during testing of the remaining flying time prediction without danger of vehicle loss.

The above discussed elements are integrated in the CAS testbed module for carrying out the experiments and research work for verification and validation of developed algorithms. For communication, testbed utilized the Live Virtual Constructive Gateway from the NASA Live Virtual Constructive-Distributed Environment (LVC-DE) Project. The LVC Gateway is a NASA tool for connecting virtual manned, virtual unmanned, and real UAVs in the field from various centers into a single real-time virtual airspace. This has been proven to be a powerful tool for training and research. We are working to extend the LVC-DE tool to better support the local airspace prognostics, safety, and efficiency research undertaken by groups at the aero centers. Our research work adds the following functionalities to LVC-DE:

- The integration of additional labs and lab resources into the LVC-DE architecture
- The addition of virtual aircraft simulated in X-Plane Flight Simulator
- Extending the LVC Gateway to handle hardware-in-the-loop airspace components
- Setup of a network resource for cloud computation supporting research and communication management

---

<sup>a</sup><http://mts.nasa.gov>



Figure 2. Edge 540 Vehicle

- Built-in support for future rapid feasibility studies and research in local airspace prognostics, control, and safety.

In addition to the experiments the test bed can be used as a research platform where researchers can develop algorithms which then can be tested in the NAS scenario under different conditions. Some of the research tests cases we propose are list below and discussed later in the paper.

- Airspace experiments with interaction between live and virtual (simulated aircraft)- Conflict avoidance, without risking loss of aircraft.
- Degraded capability/ fault injection with HITL elements without risking aircraft. Simulate conditions on HITL elements. Can even run.

### III. Testbed Architecture

A major part of the engineering in this project is the design, development, and testing of software. In this section we describe the software developed for the virtual lab infrastructure.

The Virtual Lab software was written with the following factors in mind:

**Performance** Messages had to be passed between components correctly, with very little delay to be used in HITL component driving or decision making.

**Flexibility** The infrastructure had to capable of supporting a wide range of planned distributed experiments. Any of the modules must be capable of running from any machine that has network connection.

**Ease of Use** This infrastructure needs to make experimentation easier and simpler.

**Expandability/Maintainability** The infrastructure was designed with the future in mind. As time goes on the needs of the group will change, so we have to be able to expand the capabilities of the testbed easily.

Considering these factors and the project requirements, a modular architecture was chosen. In this architecture communication is handled by a message exchange server that handles the receipt and distributions of messages from a variety of sources. For our project we chose the LVC Gateway, a NASA product that handles a variety of aerospace messages though a publisher/subscriber system.<sup>5,6</sup> The modules each manage one capability and uses a version of the client to communicate with the message exchange server.

**Table 1. Infrastructure Modules**

Module Name	Source	Description
Message Recording Module	GOTS	A message recording system designed for the LVC Gateway. This module records any messages handled by the gateway.
Display Module	Modified GOTS	A module for displaying the current aircraft states, and sensor information in a web-accessible graphical user interface (GUI). This is useful for providing state awareness to researchers. For this, the NASA Airborne Science developed Mission Tools Suite (MTS) was used. Mission Tool Suite is a collection of web-based tools to assist with the planning, operations, and overall management of airborne missions. <sup>b</sup>
Traffic Generation Module	New	Creates multiple passive virtual aircraft of a set pattern.
Playback Module	New	A configurable playback agent capable of reading a number of different playback file formats.
Research Module	New	A module for connecting research algorithms to the LVC gateway using the Generic Software Architecture for Prognostics (GSAP).
Live Aircraft Module	New	A module for connecting UAV 540 aircraft to the virtual lab.
Active Virtual Aircraft Module	COTS/GOTS Mixed	A module for connecting virtual controllable aircraft into the virtual lab.
HITL Module	New	A module for connecting hardware in the loop elements to the gateway. These elements are dynamically loaded based on the live flight information passing through the gateway.

Depending on the experiment, you can activate different modules of the infrastructure. The modules are described further in Table 1. In this table, the source is indicated to show which software products were reused. Four categories of sources are indicated: commercial off the shelf (COTS), government off the shelf (GOTS), mixed, and new. The modules marked as new are the ones that were made from scratch without any significant software reuse.

Many of the individual software components required for the virtual lab already exists and is currently being used by other projects.<sup>3</sup> When this is the case, we have tried to incorporate those software components.

### A. Virtual Lab Clients

Each module uses a virtual lab client to communicate with the gateway. Each client handles the formation, receipt, and communication of messages with the LVC Gateway. For this project we developed C, C++, LabView, and Matlab clients. Each client communicates with the LVC Gateway through a TCP connection. The clients are designed so that any number of clients of different languages can be connected to the same gateway sending and receiving messages. This is necessary to support the modular architecture critical to this application.

### B. LVC Gateway Extension-Additional Messages

The LVC Gateway is capable of managing a number of messages to convey information about an aircraft's state, and intended trajectory. These are very important for enabling airspace experiments. Absent from these messages however is a message to carry data collected from sensors internal to an aircraft, such as that from a thermocouple on a motor, or a voltage sensor on a battery. This information is critical for performing diagnostics and prognostics of aircraft components. Diagnostic and prognostic algorithms use this information to determine the degradation state of the component.

To carry data from these sensors the LVC Gateway was extended to support the SensorDataPoint message, defined below. This message was designed to be flexible, capable of carrying messages from a wide range of sensor types, and compact to limit bandwidth use. The entire message is 52 Bytes. The unit types enumeration will be expanded to support other units as needed. The added sensor message is defined below as a C++ struct.

```
#define SENSORNAMELENGTH          32

typedef enum eUnitTypes_t {
    eUNIT_TYPE_VOLTS,           /* Volts           */
    eUNIT_TYPE_AMPS,           /* Amps            */
    eUNIT_TYPE_RPM,            /* Rotations per minute */
    eUNIT_TYPE_DEG_C,          /* Degrees Celcius   */
    eUNIT_TYPE_MLS,            /* Meters per second  */
    eUNIT_TYPE_NOT_SET = -99999
} enum_UnitTypes;

typedef struct MsgSensorDataPoint_t {
    char          m_name[SENSORNAMELENGTH];
    enum_UnitTypes m_unit;
    int           m_id;
    double        m_value;
    double        m_timeCollected;
} structMsgSensorDataPointType;
```

The Sensor Data message is consumed by the Research Module, which uses that to produce prognostic results. For future versions of the infrastructure we intent to add a message to carry the prognostic results and suggested mitigation actions.

## IV. Testbed Modules

A heterogeneous network of systems with common interfaces sharing a virtual environment creates new possibilities for research. Central to this research is the integration of the live aircraft, virtual aircraft, and HITL modules. These modules make use of the features of other modules to create scientifically interesting scenarios. Experiments can make use of one or many of these.

### A. Live Aircraft

Live aircraft are field-assets which can be deployed for experimentation and are connected to the virtual lab using the Live Aircraft Module. This module is run on the embedded computer on the aircraft and communicates to the gateway through a cellular connection using the C Client. There are currently two versions of the Live Aircraft Module in use: a stand alone version and a Core Flight System (CFS) service version. CFS is a platform and project independent reusable software framework and set of reusable software applications<sup>c</sup>. The CFS version of the module plugs into CFS running onboard the aircraft.

For our lab we have a number of Edge 540 UAVs that can run this module. The Edge 540 is a small electric UAV with a wing span of approximately 100” and a weight of approximately 50 lbs. This UAV is a 33% sub-scale version of the Zivko Aeronautics Inc. Edge 540 T tandem seat aerobatic aircraft . This aircraft has been actively used by researchers at NASA LaRC to facilitate the rapid deployment and evaluation of remaining flying time prediction algorithms for electric aircraft since 2010.<sup>7</sup> Remaining flying time prediction algorithms focus on the prediction of battery charge depletion over an e-UAV flight. A lower-bound on the battery state of charge (SOC) that is considered safe for flight is set at 30% in this work. Flying the vehicle with batteries below 30% SOC is considered to be a high-risk mode of operation that violates the vehicles safe operating guidelines. Such violations of operating guidelines are referred to here as a functional failure of the vehicles mission.<sup>3</sup>

<sup>c</sup><https://cfs.gsfc.nasa.gov>



## B. Virtual Aircraft

Virtual aircraft are a simulation based source from which aircraft data is generated. Data from virtual aircraft is passed to the gateway to be consumed by other modules. Virtual aircraft enable a number of experiments that are not technically feasible with live aircraft, such as scenarios that might lead to failure. Additionally, experiments using live aircraft require significant effort and time to prepare and run, and are subject to weather. This is not true for virtual aircraft. Virtual aircraft are often used with live aircraft for complex scenarios or for testing experiments before using live aircraft.

There are two types of virtual aircraft: active aircraft which are controlled actively by a user or passive aircraft that follow some set path without knowledge of the other aircraft in the airspace. Active aircraft have the advantage of being able to react to events in the airspace.

### 1. Active Virtual Aircraft

Active virtual aircraft are simulated using X-Plane commercial flight simulator.<sup>d</sup> X-Plane provides powerful flight simulation and visuals for simulating aircraft. Pilots can drive the aircraft using a joystick or their mouse. Additional autopilot systems can pilot the aircraft using plugins. The active virtual aircraft module connects with X-Plane using the NASA developed XPlaneConnect (XPC) toolbox<sup>e</sup>. It passes the state of the piloted virtual aircraft to the gateway in the form of FlightState messages and displays the other aircraft in the experiment in X-Plane for the pilot to see. X-Plane allows for the simulation of a variety of different aircraft in varying geographies and weather conditions.

### 2. Passive Virtual Aircraft

There are two separate modules available for use in the virtual lab which can create passive virtual aircraft: A playback module, and traffic generation module.

The playback module creates virtual aircraft based on information contained in a playback file. The playback module is configurable to accept a wide variety of playback file formats. These playback files are useful for running exact scenarios or rerunning flights conducted by live aircraft. This module can also be useful for demonstrations.

The traffic generation module is used to create multiple passive virtual aircraft of a set pattern. The generated traffic is often used to add complexity to a scenario by adding traffic for a live or active virtual aircraft to avoid.

In addition to these virtual aircraft modules custom virtual aircraft can be created using the clients.

## C. HITL Module

The hardware-in-the-loop module utilizes the LabVIEW client to connect laboratory based hardware testbeds to the LVC Gateway. Through this connection, the current testbed, an electric propulsion system testbed, can be easily connected with the LVC Gateway. This electric propulsion system testbed is designed to operate an isolated electric UAV propulsion system and apply realistic flight like loads on the system as desired. The Edge 540 UAV was used as a reference aircraft from which the electric propulsion system testbed was built. The propulsion elements from the Edge 540 UAV consist of batteries, electronic speed controls, and a brush-less DC motor. The addition of hardware-in-the-loop elements such as a bench top propulsion system testbed to the virtual laboratory testbed gives researchers easy access to a scientifically relevant portion of the aircraft without the overhead and dangers encountered during actual flight. Additionally, hardware to produce mechanic or programmatically induced faults can be easily integrated without the space or weight limitations of a fuselage. This enables researchers to rapidly prototype and implement new flight hardware or aircraft control software. Finally, the bench top propulsion system testbed has comparatively fewer consequences to propulsion or energy storage system failure than the actual aircraft.

---

<sup>d</sup> <http://www.x-plane.com/desktop/home/>

<sup>e</sup> <https://github.com/nasa/XPlaneConnect>

## V. Experimental Results

### A. Milestone 1 - Virtual Aircraft

The first test of the virtual laboratory testbed included a scenario with only virtual aircraft. In this scenario a single active virtual aircraft module was operated at NASA Langley Research Center. The aircraft was controlled by an autopilot system. The Virtual Aircraft Module used a cellular connection to send flight state and trajectory intent messages to the LVC Gateway. Data was then recorded to a file by the message recording module.

The connection between the gateway and the client was then severed and reconnected to test the clients handling of communication dropouts. The data recorded was verified against the data sent. Table 2 contains some results from the experiment.

Table 2. Milestone 1 Results

Experiment Duration	22 min
Mean Latency	0.1 s
Message Drop Rate	0%
Test Result	Success

This served as an initial demonstration of the virtual aircraft capabilities.

### B. Milestone 2 - Live Aircraft with Display

The second test of the virtual lab infrastructure involved a scenario with a live aircraft. An Edge 540 UAV was piloted by a professional UAV pilot in Virginia. The Virtual Aircraft Module used a cellular connection to send flight state and trajectory intent messages to the LVC Gateway. Data was then recorded to a file by the message recording module and visualized live in the MTS display. The data recorded was verified against the data sent. Table 3 contains some results from the experiment. In this case, message drops were **do** to lost cellular connection during flight. Message latency can also be seen in Figure 3.

Table 3. Milestone 2 Results



Experiment Duration	23 mins
Mean Latency	0.1017 s
Message Drop Rate	7.3%
Test Result	Success

## VI. Discussion and Comments

The results observed during the initial milestone completion schedule look promising and we are working further to integrate the system and test it. All the hardware and software components in each of the module were tested and verified for operation for a real time scenario. Latency was a major concern due to hardware, firewalls between the two centers. During the experiments it was found that the latency is well within the tolerance limits.

The team is currently working towards Milestone 3, where the HITL testbed will be operated and loaded synchronously with the conditions onboard the Edge 540 aircraft in flight. This test will utilize the new sensor data message set and the HITL motor loading assembly within the electric propulsion system testbed. Beyond milestone 3, the team is working towards completing the research module, and creating a module to generate out-the-window visuals for live aircraft using X-Plane.

Further in future, the team will work towards integrating additional HITL modules, creating an experimental control GUI for enabling/disabling modules, and exploring experiments involving autonomous, health-aware, aircraft decision making.



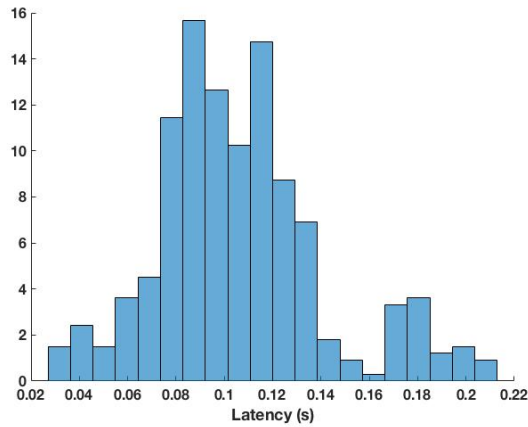


Figure 3. Histogram of Message Latency

## Acknowledgments

The team would like to thank James Murphy, Neil Otto, Srboľjub Jovic, and the rest of the LVC Gateway team for their help getting us started with their LVC Gateway software, and instruction on how to add additional message sets. Additionally, the team would like to thank Aaron Duley and the MTS Team for the generous work they put into deploying their MTS display for our project.

## References

<sup>1</sup>NASA, Ames Research Center, Moffett Field, CA 94035, *Live Virtual Constructive Distributed Environment (LVC) LVC Gateway, Gateway Toolbox*.

<sup>2</sup>NASA, Armstrong Flight Reesarch Center, Edwards, CA 93523, *Interface Control Document (ICD) for the LVC Gateway (Rev B)*.

<sup>3</sup>Bole, B., Daigle, M., and Gorospe, G., “Online Prediction of Battery Discharge and Estimation of Parasitic Loads for an Electric Aircraft,” *Second European Conference of the Prognostics and Health Management Society 2014*, July 2014, pp. 23–32.

<sup>4</sup>Bole, B., Teubert, C., Chi, Q. C., Edward, H., Vazquez, S., Goebel, K., and Vachtsevanos, G., “SIL/HIL replication of electric aircraft powertrain dynamics and inner-loop control for V&V of system health management routines,” *Annual Conference of the Prognostics and Health Management Society*, 2013.

<sup>5</sup>Murphy, J. R., Jovic, S., and Otto, N. M., “Message Latency Characterization of a Distributed Live, Virtual, Constructive Simulation Environment,” *AIAA Infotech @ Aerospace Conference*, 2015.

<sup>6</sup>Murphy, J. R. and Hoang, T., “UAS Integration in the NAS Project: Integrated Test and LVC Infrastructure,” *NASA Technical Reports*, 2015.

<sup>7</sup>Hogge, E. F., Bole, B. M., Vazquez, S. L., Celaya, J. R., Strom, T. H., Hill, B. L., Smalling, K. M., , and Quach, C. C., “Verification of a Remaining Flying Time Prediction System for Small Electric Aircraft,” *Annual Conference of the Prognostics and Health Management Society 2015*, 2015.