Proof-of-Concept Demonstrations of a Flight Adjustment Logging and Communication Network

Matthew C. Underwood[1]
NASA Langley Research Center, Hampton, VA, 23681

Daniel K. Merlino[2]
University of North Texas, Denton, TX, 76203

Lindsey M. Carboneau[3]
Florida Gulf Coast University, Fort Meyers, FL, 33965

C. Logan Wilson[4]
University of Maryland, College Park, MD, 20742

and

Andrew J. Wilder[5]
University of Kentucky, Lexington, KY, 40506

The National Airspace System is a highly complex system of systems within which a number of participants with widely varying business and operating models exist. From the airspace user's perspective, a means by which to operate flights in a more flexible and efficient manner is highly desired to meet their business objectives. From the air navigation service provider's viewpoint, there is a need for increasing the capacity of the airspace, while maintaining or increasing the levels of efficiency and safety that currently exist in order to meet the charter under which they operate. Enhancing the communication between airspace operators and users is essential in order to meet these demands. In the spring of 2015, a prototype system that implemented an airborne tool to optimize en-route flight paths for fuel and time savings was designed and tested. The system utilized in-flight Internet as a high-bandwidth data link to facilitate collaborative decision making between the flight deck and an airline dispatcher. The system was tested and demonstrated in a laboratory environment, as well as in-situ. Initial results from these tests indicate that this system is not only feasible, but could also serve as a growth path and testbed for future air traffic management concepts that rely on shared situational awareness through data exchange and electronic negotiation between multiple entities operating within the National Airspace System.

Nomenclature

AOC	=	Airline Operations Center
ASTOR	=	Aircraft Simulation for Traffic Operations Research
ATC	=	Air Traffic Control
ATM	=	Air Traffic Management
ATOS	=	Airspace and Traffic Operations Simulation

[1] Research Aerospace Engineer, Crew Systems and Aviation Operations Branch, NASA Langley Research Center, Mail Stop 152, Hampton, VA 23681-2199, AIAA Member
[2] Undergraduate Student, Computer Science and Engineering
[3] Undergraduate Student, Software Engineering and Applied Mathematics
[4] Graduate Student, Aerospace Engineering
[5] Undergraduate Student, Mechanical Engineering

DIP	=	Dispatch Interface Program
DSP	=	Database Server Program
EFB	=	Electronic Flight Bag
FALCN	=	Flight Adjustment Logging and Communications Network
FMS	=	Flight Management System
HMI	=	Human-Machine Interface
IFI	=	In-Flight Interface
IP	=	Internet Protocol
NASA	=	National Aeronautics and Space Administration
STAP	=	Simplified Text Avionics Protocol (ARINC 834)
TAP	=	Traffic Aware Planner
TASAR	=	Traffic Aware Strategic Aircrew Requests
TCP	=	Transmission Control Protocol
XML	=	eXtensible Markup Language

I. Introduction

The Network-Enabled Air Traffic Management (ATM) subproject at the National Aeronautics and Space Administration (NASA) is investigating methods to apply advanced networking concepts, modern information sharing technologies, and innovative distributed processing technologies to air traffic operations in order to improve safety, efficiency, capacity, and robustness of integrated decision making in air traffic management.

As part of this subproject, NASA is investigating several ATM applications and concepts that would benefit from enhanced data sharing; the Traffic Aware Strategic Aircrew Requests (TASAR) has been identified as the leading concept at NASA Langley. TASAR utilizes a pattern-based genetic algorithm[1] in a software application known as the Traffic Aware Planner (TAP) which is hosted on a Class 2 Electronic Flight Bag (EFB) on the flight deck of an aircraft. TAP generates a flight path that is deconflicted for traffic, weather, special use airspace, etc., while focusing on calculating a fuel, time, or combination benefit for the flight. These calculations are based on data regarding the current aircraft trajectory, weather, winds aloft, and other aeronautical information. A user interface for TAP presents three types of suggested route modifications to the flight crew—a vertical route modification only (flight level change), a horizontal route modification that conforms to named waypoints in the National Airspace System[2] only, and a combination vertical and horizontal route modification that is exclusive of both the vertical-only and horizontal-only route modifications[3]. The pilot can then choose to request one of the suggestions as a standard route modification request to air traffic control (ATC)[4].

The TASAR concept has been tested both in human-in-the-loop simulations and in a flight test[5]. Additionally, the development of TAP considered certification for operational use[6, 7]. Furthermore, analyses show the preliminary benefits that TASAR would provide for an air carrier[8]. The benefits study and discussions between the TASAR team members and several domestic airlines resulted in two airlines expressing interest in equipping their aircraft with TASAR technology.

The airline stakeholders interested in TASAR have expressed the desire to be able to display TASAR-suggested flight path modifications to dispatchers located at their airline operations centers (AOC). Current avionics and communication infrastructures do not yet transmit the information required to display this information at an AOC. However, it is hypothesized that this information can be made available to ground-based systems at an AOC by utilizing an existing data link, such as in-flight Internet.

This paper is organized as follows: Section II introduces the prototype system developed for this proof-of-concept activity. The laboratory system arrangement, including the simulation environment, the configuration of the system, the test scenario, and the results and impacts of the laboratory testing are discussed in Section III. Section IV provides a discussion regarding an in-situ proof-of-concept experiment and discusses the results of the in-situ testing of the FALCN system. Finally, conclusions are presented in Section V.

II. Prototype System

Through the use of existing in-flight Internet connectivity, a prototype system was designed and built that transmits aircraft state, intent, and route modification data to an AOC, resulting in the inclusion of dispatcher feedback in any rerouting decisions made by the flight crew. Once a dispatcher evaluates a flight crew’s request, his or her response is sent back to the flight crew via the same in-flight Internet connection. With dispatcher recommendation, the flight crew can then decide whether to request a new route clearance from ATC or stay on the current route. This system has been named the Flight Adjustment Logging and Communications Network (FALCN).

A. FALCN Component Overview

FALCN consists of four main components—an in-flight interface for the flight crew, a database server program that controls the communications between the air and the ground, a database for storing and retrieving data, and a user interface for an airline dispatcher that enables review and the ability to provide recommendations to the flight crew regarding proposed en-route flight path modifications generated by TAP. This section of the paper discusses the individual components of FALCN at a high-level. For more detailed information on each of these components, see reference 9.

1. In-Flight Interface

The In-Flight Interface (IFI) is a user interface designed to allow the flight crew to easily make a selection of a TAP generated route and send the selected route information to dispatch for review. Additionally, it gathers the aircraft state and intent data and transmits them to the Database Server Program (DSP). In the prototype system, the IFI is a standalone piece of software that interacts with the TAP interface. In future implementations of FALCN, the IFI will be fully integrated into the TAP human-machine interface (HMI).

The IFI has a login screen that allows a flight crew to input flight-specific data, including the aircraft tail number and the flight number, as well as the Internet Protocol (IP) address of the server that the DSP resides on. In future iterations of the login screen, security features (e.g., username and password authentication) will exist. Once the flight information and positive contact with the DSP are confirmed, the main IFI interface will become functional and the IFI and the TAP programs will begin to communicate.

During flight, TAP generates optimized flight route solutions once per minute when routes that improve the fuel or time efficiency are identified. Once a solution is selected by the flight crew in the TAP HMI, the IFI will populate with the information corresponding to the selected solution in TAP. At this time, the pilot has the option to send the request to the dispatcher for concurrence. The dispatcher reviews the request and sends a decision to the IFI via the DSP and in-flight Internet, where it is presented to the flight crew.

2. Dispatch Interface Program

The Dispatch Interface Program (DIP) allows the dispatcher to effectively consult with the flight crew to reach an agreeable plan of action. The DIP features several areas on the interface where information is displayed to the dispatcher. The Incoming Route Request table is a queue of all incoming route modification requests from flight crews. The route modification requests, as well as aircraft state and intent data are gathered from the DSP. When the dispatcher approves a route modification, the row is translated from the Incoming Route Request table to the Approved History table. If the dispatcher disapproves a flight crew's requested route modification, the flight crew is notified on the IFI and the request is removed from the Incoming Route Requests table.

The final aspect of the DIP is an interactive map. The map is included to improve the usability of the interface and provide information to the dispatcher as efficiently as possible. Having an interactive map allows the dispatcher to graphically visualize the proposed route quickly, without having to examine the route textually. Minimizing additional dispatcher workload is a vital aspect for adding the DIP to a dispatcher's current suite of interfaces.

3. Database Server Program

The Database Server Program is the communications hub of the FALCN system. It provides the interface through which the IFI and the DIP communicate and store data. The DSP handles clients by identifying whether the connection originates from a flight or a dispatcher and determining the specifics of the request. This requires parsing the server socket stream and developing a query to appropriately manage database interactions. Unlike the IFI and DIP, the DSP has no front end user interface and requires no direct human interaction to operate.

When an IFI client connects to DSP, the server checks the incoming data and classifies the type before inserting the information into the FALCN Database. The DSP stores data transmitted from the aircraft in the database, and any pending dispatcher responses are retrieved from the DIP and relayed back to the IFI for display to the flight crew. When a DIP client connects to DSP, the server retrieves a list of active requests and returns them to the DIP for display to the dispatcher. The server will also notify the DIP of cancelled requests and when there is an ATC decision update available for an accepted request.

4. FALCN Database

The FALCN database is designed as a relational database—every table within the database is related, or linked, to each other by a flight number key. The database was constructed to hold five tables: Aircraft State Information, Aircraft Intent Information, TAP Route Modification Requests, History, and Reasons. The Aircraft State Information table holds identification and state data information records (e.g., callsign, current altitude, current position, and airspeed). The Aircraft Intent Information table stores limited intent information in the form of the current active flight plan in the Flight Management System (FMS). The TAP Route Modification Requests table holds TAP route modification requests (e.g., off-route waypoint name, waypoint latitude and longitude, and fuel/time savings). This table updates any time a route modification request is sent from the IFI. The History table is a log of air traffic controller and dispatcher decisions with corresponding reasons describing the approval or denial of route modification requests. The History table will allow the airlines to review previous flights and analyze data trends. This table is updated whenever a decision regarding a flight route modification is made by a dispatcher or a controller. The Reasons table correlates a numeric reason value stored in the History table with a textual description stored within the Reasons table.

B. System Architecture and Data Flow

Figure 1 depicts the system architecture and data flows between components that comprise the FALCN system. As mentioned in the previous section, the IFI, DSP, DIP, and FALCN Database are the core components of the system. Data from the TAP application is also required for FALCN.

To deploy the TAP software to a variety of platforms, including EFBs running Microsoft Windows or Apple iOS, the TAP algorithmic engine was separated from the TAP HMI. This was done to allow the computationally light HMI to run on an EFB, while the computationally heavy algorithmic engine runs on a separate dedicated processor. Two eXtensible Markup Language (XML) files—one generated by the TAP algorithmic engine and the other generated by the HMI—log all data transferred between the algorithmic engine and HMI. The TAP algorithmic engine provides data including the aircraft's state and intent as well as the calculated route modification solutions and their time and fuel outcomes to the TAP HMI. The TAP algorithmic engine provides updated state data to the TAP HMI at 1 Hz, and the intent data (FMS flight plan) is updated only when the active route in the FMS changes, such as when a waypoint in the route is sequenced, or when the flight crew enters a TAP-generated flight route modification. Additionally, the TAP algorithmic engine provides updated route modification solutions once per minute to the TAP HMI, and updates the fuel and time outcomes every ten seconds.

The IFI obtains the state and intent information as it is updated and recorded in the XML file generated by the TAP HMI. It then transmits these data over a network via Transmission Control Protocol (TCP) packets to the DSP. Whenever the flight crew selects a solution on the TAP HMI, the IFI gathers the route modification information from the XML file and displays it on the FALCN IFI. When the flight crew makes a request via the IFI, it sends the route modification request via TCP packets that contain a string expressed as either named waypoints (lateral solution), a flight level (vertical solution), or a combination of the two (combination solution), as well as the fuel and time outcomes for the selected solution. The IFI also gathers and transmits the updated fuel and time outcomes as they are calculated and sent from the TAP algorithmic engine.

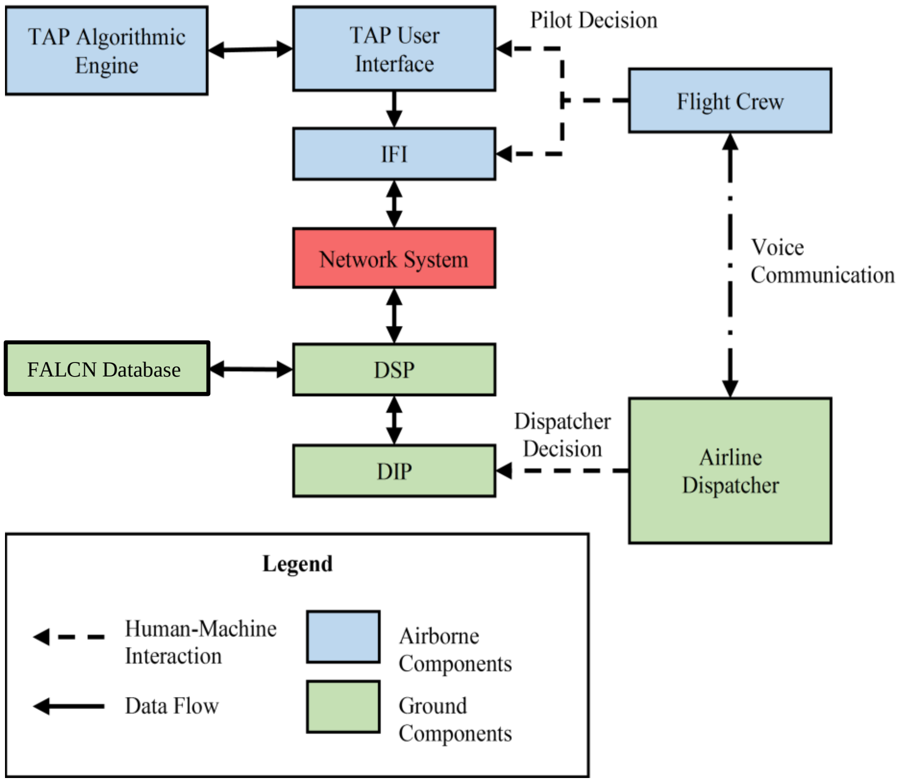

Figure 1. System Architecture and Data Flow. *The network system illustrated in this figure may be an in-flight Internet system, a terrestrial Internet system, a wireless local area network, or a hardwired local area network.*

The DSP receives these packets, parses them, stores these data in the FALCN Database, and communicates these data to the DIP so that a dispatcher can act upon the information. When a dispatcher makes a decision, the DIP transmits that data via TCP packet to the DSP, where it is stored in the FALCN Database and simultaneously sent to the IFI for the pilot to act upon.

III. Laboratory Proof-of-Concept Methodology and Results

For the initial development and demonstration of the prototype system described in Section II, a ground-based simulation and development environment was created. This section describes the test equipment, simulation capabilities, simulation configuration, and the scenario used to test the system, as well as results and impacts of the proof-of-concept demonstration.

A. Test Equipment and Simulation Capabilities

NASA’s Airspace and Traffic Operations Simulation (ATOS) is a medium-fidelity traffic simulation designed to enable researchers to develop and test vehicle-centric ConOps for NextGen implementation[10]. The fidelity and scalability achieved by ATOS results from its architecture, which links many individual aircraft simulations. To add additional aircraft to the simulation, computers are added to the network. ATOS is capable of supporting human-in-the-loop experiments or batch simulations[11].

The primary aircraft simulation within ATOS is the Aircraft Simulation for Traffic Operations Research (ASTOR), which combines a six-degree-of-freedom airframe simulation, flight management and autoflight systems, modern glass displays, and a simulation of an ARINC 429 data bus. ASTOR can be implemented on a range of workstations or simulators with various levels of fidelity, or it can be run with a pilot simulation replacing a human pilot[10].

B. Simulation Configuration

For this study, three computers were networked together to emulate the expected implementation of FALCN. All computers used in this configuration are Dell OptiPlex 745 machines with dual-core Intel Pentium D processors. Each computer has a network interface card that allows it to transmit and receive data via Ethernet. A Linksys EZXS55W network switch connects the three computers. A depiction of the configuration can be seen in Figure 2, and an image of the simulation setup can be seen in **Error! Reference source not found.**.

Computers 1 and 2 represent the in-flight components of TAP and FALCN described in Figure 1. A single-pilot ASTOR station (seen on the left in Figure 3), the ATOS Simulation Manager, and the TAP algorithmic engine exist on Computer 1. The ASTOR used in this study has a math model that is representative of a Boeing 757, and glass cockpit displays consistent with a Boeing 777. The ATOS Simulation Manager controls the execution of the simulation. The TAP algorithmic engine communicates with the ASTOR through the simulated ARINC 429 data bus. A Data Concentrator Emulator converts the ARINC 429 words into a Simple Text Avionics Protocol (STAP) feed following the standards set forth in

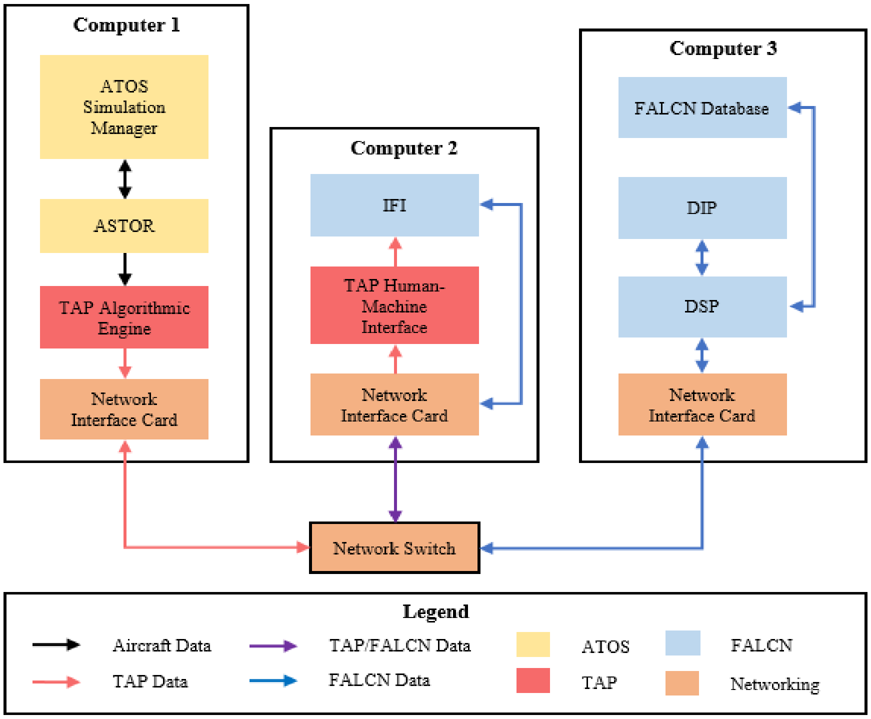

Figure 2. FALCN Simulation Environment. *Computers 1 and 2 represent the airborne capabilities, and computer 3 represents the ground-based capabilities.*

ARINC 834 before the aircraft systems data is transmitted to the TAP algorithmic engine. The algorithmic engine then sends this information over the network to Computer 2. The second machine hosts the TAP user interface, seen in the left-center of Figure 3, and the IFI, seen in the right-center of Figure 3.

Finally, Computer 3 represents the ground-based components of the FALCN system (seen in Figure 1) that would be located at an airline's flight operation center. The third computer hosts the DSP and DIP. The DIP is shown on the right of Figure 3.

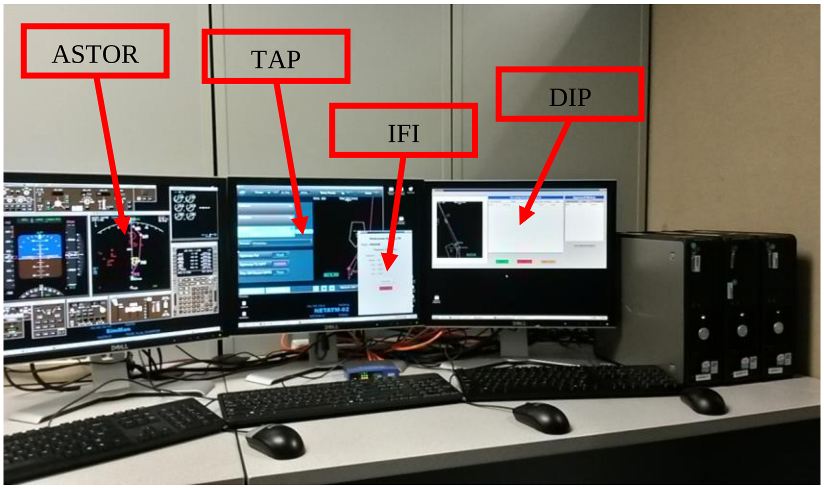

Figure 3. Simulation Environment in Laboratory. *The ASTOR, TAP HMI, IFI, and DIP are shown in operation. Not shown is the ATOS Simulation Manager. The TAP algorithmic engine and the DSP have no user interfaces.*

C. Test Scenario

To develop the FALCN software and assess its integration with the TAP software, a test scenario was utilized. The scenario featured a flight from John F. Kennedy International airport in New York City to Denver International Airport. The scenario initialized over Pennsylvania, just west of Philadelphia, at FL300. The flight route was sub-optimal due to impacts from a region of Special Use Airspace and convective weather.

TAP was connected to the aircraft systems and provided optimized route modifications to the flight crew. As well, FALCN was connected to TAP and offered the opportunity for the flight crew to contact dispatch. On-the-ground systems (the DSP, DIP, and FALCN Database) were connected to a local network and provided the dispatcher with the TAP route modifications that were sent from the IFI.

D. Laboratory Demonstration Results and Impacts

In April 2015, the system was demonstrated to an internal group of NASA civil servants and contractors who are familiar with the TASAR concept of utilizing a high bandwidth data link, such as in-flight Internet, to augment current communication infrastructures that provide data to airborne ATM concepts and applications. The system has also been demonstrated to external stakeholders, including in-flight Internet service providers and ATM concept developers.

1. Results

The demonstration was successful and accomplished the goals set forth at the start of the activity. Data was successfully communicated between TAP, the IFI, and ground-based FALCN subsystems. A worst-case bandwidth requirement for the data being communicated by FALCN of 24 kilobits per second was derived, which is roughly forty percent of the maximum dial-up Internet bandwidth. The feasibility of in-flight Internet providing enough bandwidth to support this concept was confirmed. Round-trip latency was essentially null due to the hardwired local area network.

2. Impacts

After the demonstration was given, the project leaders of TASAR/TAP provided valuable constructive criticism and advice for future development. Among these suggestions were changing the verbiage on both the IFI and the DIP to more accurately reflect the roles and responsibilities of the dispatcher, refining the procedure for using FALCN, and finally, integrating the IFI with the TAP HMI and adding logic to notify the flight crew when a dispatcher's concurrence was necessary. These suggestions were noted and will eventually be features of a new version of FALCN.

Based on the success of the laboratory testing, it was decided to test the FALCN system in-situ. Data such as round-trip latency were required to further validate the feasibility of using FALCN as well as to adjust the procedures for using FALCN, and that data could only be gathered in-situ.

IV. In-situ Proof-of-Concept Methodology and Results

To validate the design of the FALCN system, an in-situ flight testing activity was designed and implemented. In addition, software was created to gather data on round trip (air-ground-air) latency and connectivity. This section describes the test design, modifications made to the laboratory proof-of-concept simulation capabilities, the in-situ system configuration, and results and impacts of the in-situ demonstration.

A. In-Situ Test Design

To minimize cost and maximize the amount of data gathered, a test was designed to utilize existing in-flight Internet communication systems on commercial flights. For this proof-of-concept activity, researchers from NASA Langley Research Center ran the FALCN IFI software on laptops on-board commercial flights. The FALCN DSP and DIP were run on servers at an on-site contractor's facility. For the test to be considered successful, data would be transmitted autonomously from the IFI programs running on the laptops onboard the aircraft, through the in-flight Internet, to the DSP on the ground. The DSP then transfers the data to the DIP on the ground, a human would respond to the route modification request through DIP, and the response would be transmitted through the DSP to the IFI programs on the laptops onboard the aircraft.

B. Modifications from Simulation Capabilities

To test the system in-situ, the FALCN software was modified. The modifications included updating the software from Windows XP to Windows 7, creating a partially-automated version of FALCN, and employing configuration files. Additionally, a means by which to obtain the data transferred from the TAP algorithmic engine to the TAP HMI without running TAP was required. Finally, software that gathers data on the round-trip latency and drop-out rate of in-flight Internet communications was required.

The re-implementation of FALCN for the Windows 7 operating system involved updating the libraries for the communications and for the HMIs, and upgrading the database software. The necessity for automating the FALCN functionalities stemmed from the in-situ demonstration and test. It was desired for the IFI to autonomously transmit in-flight data to a ground server. These modifications included delegating the data parser in the IFI to search through the XML code for a new solutions update from TAP and to always select the combination vertical and lateral route solution. Additionally, logic was designed and implemented in the IFI and DSP to change certain Boolean flags to allow for automated transmission and reception of data by the IFI. These modifications altered the IFI program's behavior to search for a new solution update from TAP, to always select the combination vertical and lateral route modification solution, and to automatically transmit that solution information to the DSP.

Additionally, the use of configuration files for setting initial conditions was required due to the new automated functionality. This would enable researchers to input specific data requirements, such as flight number, aircraft number, and IP address that would have otherwise been input on the login screen of the IFI.

For TAP not to be required for the in-situ proof-of-concept demonstration, and since none of the researcher's laptop computers are connected to flight systems, a means by which a TAP-generated XML file could be created was required. An XML replayer script was written that accepts as input a saved XML file from the laboratory setup, and outputs a duplicate file in the same manner that TAP would create the file.

Finally, software was created to gather the round-trip latency data. The software utilized the "Ping" function, which sends a small packet (64 bytes) of data to a particular IP address. The computer that sends the packet waits for a return packet, then calculates the time difference from when the packet was sent by the host computer and when it was received by the host computer. The IP address pinged by the computer on-board the aircraft (the researcher's laptop) was the IP address of a server located in Hampton, VA.

C. System Configuration

For the in-situ FALCN system configuration, three computers are required. Figure 4 shows a graphical depiction of the system configuration. The first computer, an HP laptop running Windows 7, hosts the TAP XML file replayer and the IFI. This computer is connected to the in-flight Internet onboard an aircraft. The IFI autonomously sends and receives data from the DSP, as mentioned in Part B.

Computer 2, a ground-based Linux server, runs the DSP and hosts the FALCN Database. This server exists in a perimeter network that is external-facing to the Internet. The second computer communicates with the IFI on-board the aircraft and the DIP, which is hosted on the third machine.

The third computer, Computer 3, hosts the DIP and communicates only with the DSP. In this configuration, the DIP was not automated and required a human to operate it.

This configuration more resembles the end-user implementation at an AOC, where each aircraft has an instance of the IFI communicating with TAP (represented by Computer 1 in Figure 4), an external-facing server hosting the DSP (Computer 2 in Figure 4), and an instance of the DIP for each dispatcher running on his or her local machine (Computer 3 in Figure 4). Computers 2 and 3 were networked via a wireless local area network, however, in some implementations, depending on the network configuration of the AOC, this network may be implemented through a hardwired local area network.

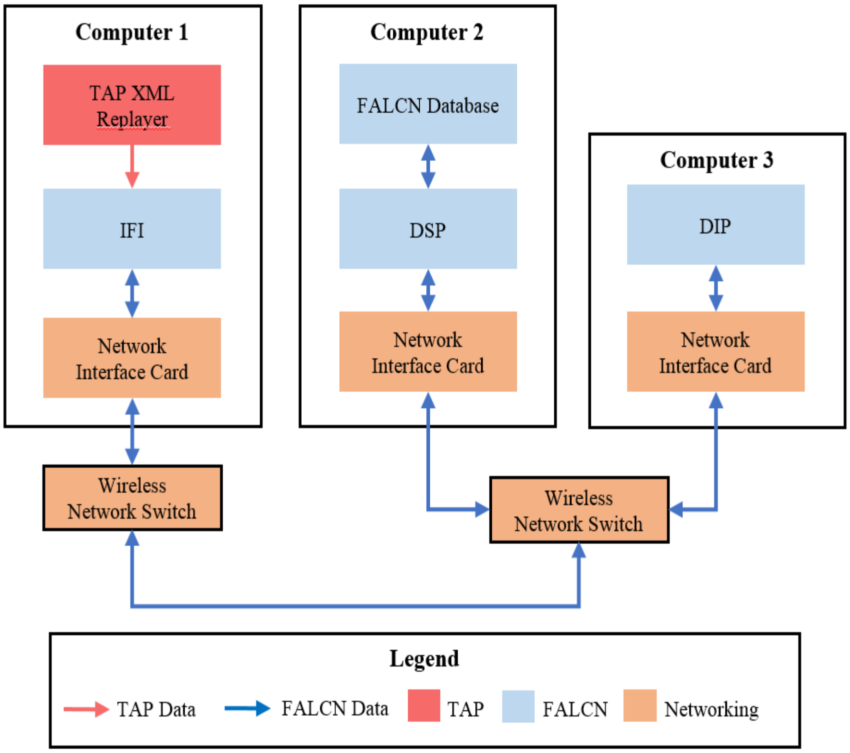

Figure 4. Configuration for In-situ Demonstration. *Computer 1 represents the in-flight component of the FALCN system. Computer 2 was an external-facing server at an on-site contractor's facility, and Computer 3 was a computer located at the on-site contractor's facility.*

D. In-situ Demonstration Results

The in-situ flight demonstration activity was a success. Ten test flights were conducted during this activity. The first set of test flights included round-trip flights from Norfolk, VA to Dallas, TX and Atlanta, GA, as well as a round trip flight from Richmond, VA to Boston, MA. A second set of test flights were used to gather additional latency data and determine the robustness of the connectivity when using air-to-ground Internet. The second set of flights included flights from Norfolk, VA to Atlanta, GA, from Atlanta, GA to Oklahoma City, OK, and return flights on those same legs. During the flight test activities, over 25,000 data points were obtained for latency values and 3,445 data points were gathered in order to determine the robustness of the Internet connectivity.

During the outbound flights of the first set of testing, FALCN was successfully able to transmit the pre-recorded data from the researcher laptop located on the aircraft, through the in-flight Internet to the DSP and ultimately to the DIP. Additionally, a developer who was operating the DIP was able to respond to the requests that were sent from the researchers on the aircraft and the researchers were able to see the responses on the aircraft. The operation of FALCN was successful, and the feasibility of using this system in-flight was proven.

The software that gathered latency data was designed in order to provide valuable insights into impediments that may limit the operation of FALCN in-situ. If communication latencies were on the order of 30 seconds or greater, then it was assumed that using FALCN in operations and acting on data that was transmitted through FALCN would be infeasible. However, of all the latency values that were gathered in all flights (25018 data points), 5.47% of the recorded latencies were over 1 second, 0.474% were over 2 seconds, and 0.0518% were over 3 seconds, with the maximum latency being 3.757 seconds. The number of large round-trip latency values (large defined as round-trip latency values greater than 1 second) was found to be inconsequential to the use of FALCN in-situ.

The average latency value for all data gathered during all flights was 476.8 ms, with a standard deviation of 308.1 ms. The median round-trip latency value was 396 ms, the maximum was 3.757 seconds, and the minimum latency value was 112 ms.

To refine the procedures in simulation rather than in-situ, a statistical model of in-flight Internet performance that accurately represents the natural behaviors that are expected when using in-flight Internet for ATM operations was created. To create the model, several distributions were fit to the data gathered during the flight test. The data, when presented in the form of a histogram, looked like a lognormal distribution. However, literature[12, 13] suggested that network latency data may align with either a lognormal, Weibull, Rayleigh, or Gamma distribution. Analysis, as seen in Figure 5, revealed that these data were best fit to a lognormal distribution with a mu parameter of 6.0075 and a sigma parameter of 0.54912.

Finally, during the first series of flight tests, the researchers conducting the tests noted that the connectivity on board the aircraft was intermittent at times. Based on these observations, the software that gathered latency data was modified to gather data on the connection status of the in-flight Internet. The data gathered during the second series of flights showed that there was connectivity for 93.3% of the time that the software was in use (at altitudes above 10,000 ft, in accordance with FAA regulations for operating personal electronic devices on an aircraft). As well, as Figure 6 presents, when the connectivity was interrupted, the median interruption was 12 seconds. The minimum connectivity drop-out time was 7 seconds, and the maximum time that connectivity was lost was approximately 129 seconds. Eighteen data points out of 136 (denoted by the red crosses in Figure 6) represent those connectivity drop out times greater than approximately 30 seconds.

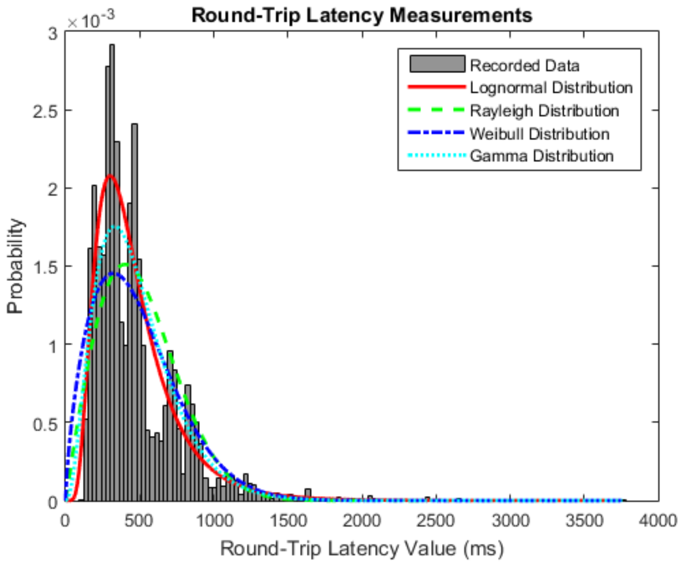

Figure 5: Histogram depicting round-trip latency value versus the probability of it occurring, as well as distributions fit to these data. *These data were best fit to a lognormal distribution, followed by a Gamma distribution.*

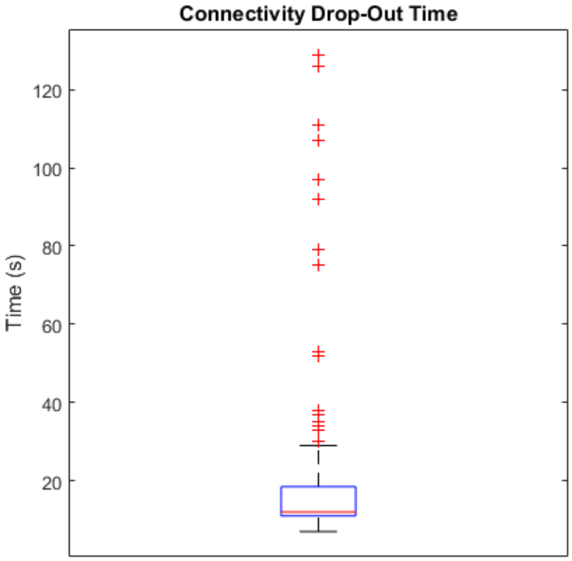

Figure 6: Connectivity Drop-Out Time. *Boxplot depicting the duration of connectivity loss during in-situ testing.*

V. Conclusion

This paper describes the proof-of-concept demonstrations and the feasibility of the FALCN system within laboratory and in-situ testing. The laboratory demonstration was confirmed with a worst-case bandwidth of 24 kilobits and a round-trip latency of close to null, with the local area network being hardwired as per system configuration design. To further prove feasibility, FALCN software was tested in-situ using commercially available in-flight Internet on ten commercial flights gathering over 25,000 latency value and 3,445 Internet connectivity data points. All data points averaged a round-trip latency time of 476.8 ms and returned a maximum time of 3.757 seconds and a minimum time of 112 ms. Furthermore, the connections proved to be robust and reliable approximately 93.3% of the time that the software was in use, with the largest connectivity drop-out time being 129 seconds. FALCN successfully transmitted and stored aircraft state, intent, and route modification requests via commercially available in-flight Internet, as well as demonstrated capabilities that in the future, may increase effective collaboration and communication between airline dispatchers, flight crews, and air traffic controllers.

Acknowledgments

The authors wish to thank Dr. Elizabeth Ward for assembling and advising the Aeronautics Academy; Mike Palmer, Steve Kohler, Joel Brockman, Ram Gubbala, Ed Scearce, Chad Chapman, and other members of the Airspace and Traffic Operations Simulation development team and Air Traffic Operations Lab operations team for providing software support and simulation hardware; Joshua Yabut for serving as a subject matter expert on cybersecurity issues; Lisa Rippy, Steve Velotas, Stella Harrison, Neil O'Connor, David Wing, Nelson Guerreiro, Gary Lohr, Roy Roper, and the Crew Systems and Aviation Operations Branch for interest, oversight, and feedback on the demonstration activity; as well as all others who contributed to this endeavor, including Christopher Newport University, Virginia Space Grant, Florida Space Grant, and Kentucky Space Grant Consortia.

This work was supported by the NASA SMART-NAS Project, Net-Enabled Air Traffic Management Subproject. The support and guidance provided by the subproject manager, Deepak Kulkarni, the associate project manager, John Koelling, and the project manager, Shon Grabbe, are greatly appreciated.

References

[1] Vivona, R. A., Karr, D. A., & Roscoe, D. A. (2006). Pattern-Based Genetic Algorithm for Airborne Conflict Resolution. AIAA Guidance, Navigation, and Control Conference and Exhibit. Keystone: AIAA.

[2] Karr, D. A., Vivona, R. A., & Wing, D. J. (2013). Costs of Limiting Route Optimization to Published Waypoints in the Traffic Aware Planner. AIAA Guidance, Navigation, and Control (GNC) Conference. Boston: AIAA.

[3] Ballin, M. G., & Wing, D. J. (2012). Traffic Aware Strategic Aircrew Requests (TASAR). AIAA Aviation Technology, Integration, and Operations Conference. Indianapolis: AIAA.

[4] Henderson, J. (2013). Traffic Aware Strategic Aircrew Requests (TASAR) Concept of Operations. Hampton: NASA/CR-2013-218001.

[5] Maris, J. M., Haynes, M. A., Wing, D. J., Burke, K. A., Henderson, J., & Woods, S. E. (2014). Traffic Aware Planner (TAP) Flight Evaluation. AIAA Aviation 2014. Atlanta: AIAA.

[6] Koczo, S. (2013). Analysis of Operational Hazards and Safety Requirements for Traffic Aware Strategic Aircrew Requests. Hampton: NASA/CR-2013-218002.

[7] Wing, D. J., Ballin, M. G., Koczo, S., Vivona, R. A., & Henderson, J. M. (2013). Developing an Onboard Traffic-Aware Flight Optimization Capability for Near-Term Low-Cost Implementation. AIAA Aviation Technology, Integration, and Operations (ATIO) Conference. Los Angeles: AIAA.

[8] Henderson, J., Idris, H., & Wing, D. A. (2012). Preliminary Benefits Assessment of Traffic Aware Strategic Aircrew Requests (TASAR). AIAA Aviation Technology, Integration, and Operations (ATIO) Conference. Indianapolis: AIAA.

[9] Merlino, D. K., Wilson, C. L., Carboneau, L., Wilder, A., & Underwood, M. C. (2016). Design and Development of a Flight Route Modification Logging and Communication Network. AIAA Aviation Technology, Integration, and Operations (ATIO) Conference. Washington: AIAA. (Submitted for publication)

[10] D. Finkelsztein, T. Lung, R. A. Vivona, J. Bunnell, D. Mielke and W. Chung, "Airspace and Traffic Operations Simulation for Distributed ATM Research and Development," AIAA-2005-6488, 2005.

[11] M. Peters and M. Ballin, "A Multi-Operator Simulation for Investigation of Distributed Air Traffic Management Concepts," AIAA-2002-4596, 2002.

[12] M. Karakas, "Determination of Network Delay Distribution over the Internet," Middle East Technical University, Ankara, 2003.

[13] M. Kwon, "A tutorial on Network Latency and its Measurement," Rochester Institute of Technology, Rochester, 2015.