2014 International Conference on Unmanned Aircraft Systems (ICUAS)
May 27-30, 2014. Orlando, FL, USA

Long-Range Communication Framework for Multi-agent Autonomous UAVs

Elchin Mammadov
School of Electrical Engineering and Computer Science
University of Ottawa
Email: emamm077@uOttawa.ca

Wail Gueaieb
School of Electrical Engineering and Computer Science
University of Ottawa
Email: wgueaieb@eecs.uOttawa.ca

Abstract—**Multiple unmanned aerial vehicles (UAVs) with inter-UAV communication capabilities can be used to extend the communication range with the ground control station (GCS). Researchers from the Mechanical and Electrical Engineering at the University of Ottawa have developed a new dirigible autonomous UAV with a flight duration of 24+ hrs, a limitied payload of 1 kg for electronics, and requiring a communication range of 1-10 kilometres. To support this requirement a new communication framework was introduced and implemented based on the ad hoc network concepts. With one radio module per dirigible the designed and developed wireless interface allows any UAV or the GCS to exchange flight control commands, telemetry data, and aerial photos. We made use of the advanced networking tools of the Digi's 9XTend$^{\text{TM}}$ radio modules to develop route tracing, traffic prioritization, and minimizing self-interference between simultaneous transmissions. Initial test results showed that without any data flow control in the network, packets can be received in the wrong order following different routes and cause errors in the transmission of photos or recorded video. This issue was resolved through acknowledgements to control the flow of packets. Using radios with half-wavelength dipole antennas we were able to achieve a one-hop range of up to 5 km with the radio-frequency line-of-sight.**

I. INTRODUCTION

A survey of fixed-wing and rotary-wing civilian UAVs in [1] reveals that their flight duration is 1 hr at most. In the context of this paper, a civilian UAV refers to an unmanned flying platform with a payload of up to 1 kg (payload is directly related to cost) and using Industry, Scientific, and Medical (ISM) bands for communication. The short flight duration is another reason why civilian UAVs do not require long range communication. For communication over a limited range and a high data rate most UAVs use the IEEE 802.11 a/b/g/n standard based Wi-Fi technology [2][3]. The range of Wi-Fi is limited to 70 and 300 m in an indoor and outdoor environment, respectively. To increase the range up to 1-1.5 km a high-gain directional antenna can be used, but in a mobile environment it would require a tracking antenna. The other solution used for increasing the range between a UAV and the GCS is using the 900 MHz with high-gain antennas. Some of the developed civilian UAVs use both bands and switch between them depending on the distance of UAV from GCS.

The possibilities of using a lightweight and a small UAV is limited by fewer resources such as a small payload, a short battery life, and a limited power in wireless communication [4]. These limitations can be compensated for by utilizing several UAVs or several radio nodes as a group. In addition, a group of effectively communicating light and small UAVs have useful properties. Based on the circumstances taking places worldwide, it is expected that the near future applications of large UAVs will require multi-agent systems [5]. It will be simpler and cheaper to conduct the refinement of the emerging multi-agent control algorithms on a smaller scale UAVs first. Reliable wireless connections among multiple agents can be used to extend the range of the communication compared to a single UAV's capability.

Most of the modern multiple-UAV control systems lack autonomous decision making to effectively reach a solution for a given task [6][7]. The basis of the effective autonomous decision making in a group of UAVs is the effective inter-UAV communication. In a review of distributed autonomous robots, Parker articulated the importance of implementing and maintaining a reliable multi-robot communication [8]. A simulation study concluded that sharing just a small amount of data (e.g. for our system it is the position of each dirigible UAV) among multiple robots can make their application more effective [9].

The main objective of this paper is to describe a newly designed and developed communication framework that supports multi-agent robotics systems. Initially being designed for a dirigible UAV, called Sensor Platform for Observation and Tracking (Figure 1), the framework was developed as a test bench for any multi-agent system.

A single uOttawa SPOT is capable of flying for 24+ hrs with a cost of $5000 CAD and requires a long-range communication in the range of 1-10 kilometres [1]. Multiple SPOTs can be utilized for an effective coverage of a search and rescue area. The initial design presumed that after deploying a single SPOT in the air, 2-5 radio nodes would be spread across the area. As the SPOT flies over a given area it would communicate to the closest radio node with the strongest received signal strength (RSS). The final design was developed supporting communication between 2-5 airborne SPOTs.

The next sections of this paper are organized as follows: Section II provides the review of the related work and different implementations of ad hoc networks for civilian

978-1-4799-2376-2/14/$31.00 ©2014 IEEE

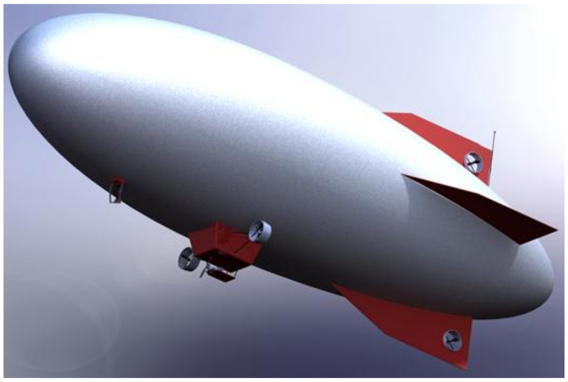

Fig. 1. SPOT UAV dirigible

UAV applications. Section III provides an overview of the proposed framework's hardware-software configuration and data processing protocols. Section IV presents the analysis of the results collected from real world test scenarios. Finally, Section V draws the conclusions and discusses possible future improvements which can be made to the framework.

II. RELATED STUDIES

Applications of civilian UAVs are vast ranging from agriculture and construction to the oil and gas sector and the security sector [10][11][12][13]. Among these applications, Sarris in [14] mentions using civilian UAVs as a communication satellites, connecting locations over a long-range. Limited in payload due to their size, each civil UAV is designed for a specific task. In cooperation these civil UAVs can accomplish a wide range of multiple tasks in a short period of time. However, the main research focus so far is to develop and improve various control algorithms for the civil UAVs. These algorithms simply assume the inter-UAV communication to be granted. The communication may not be important for simulations or close-range operation because technology such as Wi-Fi provide quick, secure, and reliable links. Wi-Fi technology is good for operation within 50-100 meters range with omni-directional antennas and 800-1000 meters with high-gain directional antennas. However, it can not provide long-range communication in the order of several kilometres.

Limited transmit power and the fixed frequency bands are the main limiting factors in achieving beyond line-of-sight communication links. Commercial solutions offer satellite communication equipment. Satellites are used as relay stations; transmitting data between two points on the ground. This allows achieving beyond LOS communication between a GCS and a UAV. Alternative to satellite communication, beyond LOS communication can be achieved using multiple UAVs for relaying data among themselves. Multiple UAVs cooperate to relay the transmitted data to the destination using 900 MHz or 2.4 GHz links. Such networks can be referred to as multi-hop ad hoc networks, wireless sensor networks, or wireless embedded networks. The achievable distance depends on the ad hoc routing protocols used, the power management configuration, and type of antennas used on the individual wireless nodes.

The need to have cooperating UAVs mainly stems from search and rescue missions. These situations have the requirement to cover multiple geographical ground and aerial regions simultaneously. For example, Rohde et al. in [15] have developed a system for optimized spatial sensor coverage using an autonomous swarm of Micro UAVs (MAV). Their system relies on the RSS and GPS data being shared between neighbouring nodes and they point out that the existing control system and the communication methods are still not working together effectively and more research is needed. Goddemeier et al. in [16] have proposed a distributed algorithm that achieves fast exploration of a given area and also relies on inter-UAV connectivity. Vachtsevanos et al. in [17] have summarized a list of data generally required to be shared among multiple UAVs in their control systems. This also opens possibility of collaboration between ground and aerial robots. An excellent example of search and rescue is the earthquake that happened in the Spring of 2011 in Japan. As Levy summarizes in [18], the rescue teams experienced problems with communication between air and ground autonomous vehicles and emphasizes the requirement for more research in this area. The article mentions the successful application of the Honeywell's T-hawk MAV and Helipse's unmanned helicopters, which use military-level radios manufactured by Cobham for the transmission of data. The article then concludes with expressing a need for having a relay communication between multiple non-military autonomous vehicles. In this article we attempt to address this problem by building a communication framework which can help multiple robots collaborate using a multi-hop ad hoc network with routing capabilities.

The inter-UAV communication challenges are described in more detail by Ryan et al. in [19] where the existing control algorithms for the non-military UAVs are summarized. Authors point out that the wireless communication between individual UAVs remain an issue and a verified implementation is yet to be found. The requirements for the UAV-to-UAV links are:

- Software interface for the flight controller in each UAV
- Reliable transmission with low transmit power and omni-directional antennas
- Using variable size data packets for transmission of telemetry and image data
- Maintaining link reliability as the network scales with the number of UAVs in the group

Matczynski in [20] proposed an embedded software architecture for use in multiple UAV control. The author points out that the main design limitation came from the network's star topology of the Piccolo autopilot communication; all the messages must be routed through a central node that manages the mission from the ground. A suggestion was made for future development to have direct links among UAVs, eliminating the need of routing through a central node. A similar communication system for multiple UAVs has been proposed by Chen et al [21]. Using radio modems at 900 MHz the authors introduce a communication method which attempts to resolve issues of data collision and timing control when multiple UAVs are transmitting. In their design the central node traverses a list of UAVs and allows each one to transmit one packet within 100 milliseconds on the network. Every node that wants to transmit information must wait for its turn and consecutive packets are delayed for at least 100 milliseconds. In both designs, [20] and [21], data collision is avoided but the speed of communication is limited by the central node that controls all the traffic. The goal for our communication framework is to allow the parallel transmission of data among multiple pairs of dirigible UAVs in a decentralized network.

The performance of the cellular links in the UAV environment has been explored by Goddemeier et al. [22]. The authors performed field-trials using an aerostat with sensors and a fixed-wing UAV to measure cellular link performance at different altitudes. Their analysis is based on RSS values. The article describes changes in the RSS values as the UAV's height changes. The authors show that as the UAV flies higher the RSS values decrease and the maximum height for a reliable link was determined to be around 500 m. In a cellular network, the transmit antennas used on towers are directional and point toward the ground; therefore, as the cellular receiver is raised it goes outside of the coverage area. This result reveals how cellular technologies can be good candidates for the inter-UAV communication in altitudes up to 500 meters. Wzorek et al. in [23] demonstrated a control of a Yamaha RMAX helicopter through a mobile phone application. A helicopter was operated at an altitude of 25-35 meters, which is in harmony with the cellular link measurements in [22]. It is important to point out that the above mentioned multi-UAV communication techniques rely on the availability of cellular service in their area.

The analyses in [22], [23] indicate that a wireless link's performance, such as transmission delay and successful delivery rate, depend on the RSS. In some situations there is no accurate way of measuring the delay or the success rate; therefore it is sufficient to measure the RSS values to estimate the performance of the link. In addition, the ad hoc networks have limited bandwidth and may not have a radio line-of-sight. As a result, a centralized routing hierarchy with preassigned routing functionality nodes are ineffective. All of the nodes should be able to detect their neighbours and perform the routing, which provides alternative propagation paths, hence increasing the reliability of data transmission [24].

Currently there are two accepted standards which can be used for implementing a mesh network. The 802.11 standard mainly used in Wi-Fi technology or the 802.15.4 standard commonly used for building low-powered personal wireless networks with ZigBee. Using Wi-Fi multiple peer-to-peer ad hoc connections can be established (star topology), but there is no support for routing between these connections. An additional software is required to resolve addressing for routing packets from one ad hoc link to another. Authors of Ad Hoc UAV Ground Network (AUGNet)[26] implemented a custom Dynamic Source Routing (DSR) by using a software called "The Click Modular Router" from [25]. Such mesh design can be summarized in terms of implemented layers and their counterparts from OSI model, as shown on Figure 2.

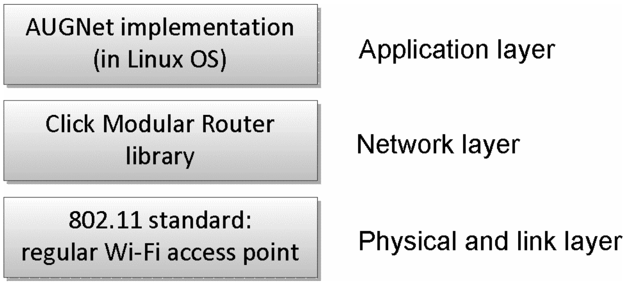

Fig. 2. AUGNet Layers

AUGNet demonstrates that the range of communication between a source and destination is increased from 1 to 3 km when relaying data through a UAV that provides better line-of-sight, hence higher throughput and lower packet losses, compared to a scenario without a UAV (routing through ground nodes only). Wirtz et al. demonstrated a simpler approach to creating Wi-Fi based centralized mesh in [27]. Rather than building a custom routing protocol the authors used Wi-Fi cards with special drivers and software from Ubuntu Linux that allowed them to setup a mesh network. The main limiting factor in 802.11 based routing implementations is the modulation techniques which are power inefficient, but provide high data rates for a fixed bandwidth. Since the main goal of our design is reliably transmitting data in a decentralized mesh and not the high data rate, a proprietary routing protocol based on 802.15.4 standard is used.

III. PROPOSED FRAMEWORK AND METHODOLOGY

The proposed communication framework allows multiple UAVs to exchange data and extend the communication range with the GCS by transmitting data in a daisy chain formation. The framework can work in any formation and data packets do not have to go through each node. We are assuming that a chain formation provides the longest communication range. The framework was designed to satisfy requirements of the SPOT, but the research has shown that it can be used for testing new multi-agent systems and emerging routing techniques. Our design incorporates the hardware and the software required for establishing a long-range wireless communication between all the SPOTs and the GCS, independent of the surrounding infrastructure. The framework is not designed to operate as a stand-alone program. It is implemented as an Application Programming Interface (API) for the SPOT's flight controller. Figure 3 illustrates the software layers. The API was developed for the Linux operating system, therefore, it uses the C POSIX library for accessing the serial ports of the computer, handling data files, and the system's time functions for the testing purposes. The design for the highlighted layer and the implementation is presented in the following sections. The main task of the framework is extending the radio control of a civilian UAV by relaying the commands in an ad hoc network using a novel mesh routing protocol called DigiMesh^{TM}.

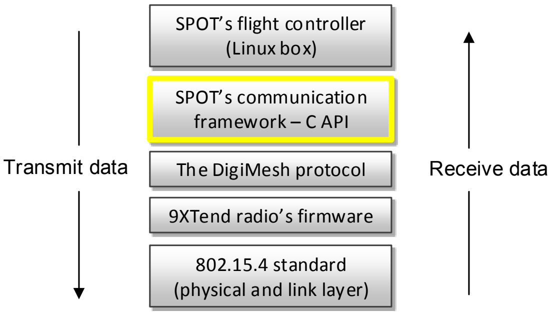

Fig. 3. SPOT Network Layers

DigiMesh is a propriety protocol, developed and maintained by Digi International. The physical and link layers are implemented according to the 802.15.4 standard. The topology of the network using the DigiMesh protocol is a true mesh. True mesh means that any node in the network can detect and route packets to any of the neighbour nodes within its communication range. DigiMesh uses its own version of a reactive protocol to achieve this kind of routing. We have used Digi 9XTend radios, which provide the implementation of the bottom three layers illustrated on Figure 3. The software interface between the 9XTend radios and the framework's API provides tools for the flight controller to monitor the state of communication by comparing the RSS against the radio's sensitivity, changing the transmit power level, and prioritizing the transmission without the operator's input. The distinguishing feature of this configuration is autonomous control of the radios by the flight controller. Both the transmission of data to other SPOTs and configuration of the radio is performed through radio packets. The framework's API interfaces with the radios using predetermined packet structure (developed by Digi) shown on Figure 5.

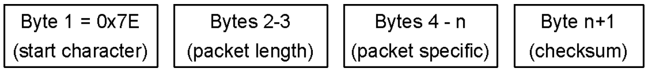

Fig. 5. DigiMesh radio packets

The packet specific bytes change depending on the type of message sent/received from the radio. The following types of messages are used by the framework: radio configuration commands, transmit SPOT's data, acknowledgement for a configuration change, status of the transmitted packets, and the received data. The maximum payload of 256 bytes per packet is enough to send all the sensor data from a SPOT to any other node using one packet. Currently all of the on-board sensors required a minimum payload of 70-100 bytes. The serial interface baud rates supported by the radio range from 9600 to 230400 baud per second but the over-the-air rate is fixed to 115200 baud/s. Therefore, the serial interface rate is set to 115200 baud/s. For the over-the-air transmission the radio uses binary-FSK modulation, which transmits one bit per one baud, hence the effective data transmission rate in the framework is 115200 bps. If the serial interface rate is not equal to the over-the-air rate, a data flow control must be implemented over the serial interface to avoid loss of data. The 9XTend radio has an internal 2.1 kilobyte data buffer, which overflows if either of the transmission rates is not controlled.

The framework's API is also designed to perform data management. In a mesh network a single receiver can be processing multiple unrelated data. For example the first packet carries image data, the next packet carries flight control commands (in addition each can be from different transmitter). We have design and developed a simple protocol at the receiver to identify type of data in each packet at the transmitter. When transmitter puts the data into the payload portion of the packet it uses the first byte as a flag to describe the data that follows it. Based on this flag the receiver is able to autonomously process multiple different streams. The flowchart on Figure 4 describes the data processing algorithm implemented for the receiver.

IV. PERFORMANCE EVALUATION AND RESULTS

In order to verify the overall expected performance of the framework several experiments have been carried out. The tests were conducted in indoor and outdoor real world environments. Three 9XTend 900 MHz radios supporting the DigiMesh protocol were used to create a small mesh network for testing. The completed tests were designed to evaluate the following aspects of the framework:

- Effective throughput measurements for single and two hops
- Self-healing links - best route selection
- Reliability of connections during image transmission
- Data collision and self-interference handling

A. General Test Procedure

Our experimental testbed configuration consists of 3 9XTend radios, running firmware with the DigiMesh routing. Each radio was connected to a laptop, which runs a Linux OS. The actual on-board computer of SPOT will eventually be running TinyOS version of Linux. The photos on Figure 6 show the three radio nodes during tests. The radio nodes on Figure 6(a) and (c) were used as a transmitter/receiver while the node on 6(b) was used as an intermediate router.

In order to test the image and text-based command transmission, additional functions such as timing file transfer and route tracing have been added to measure the framework's overall performance. A small test program was written to

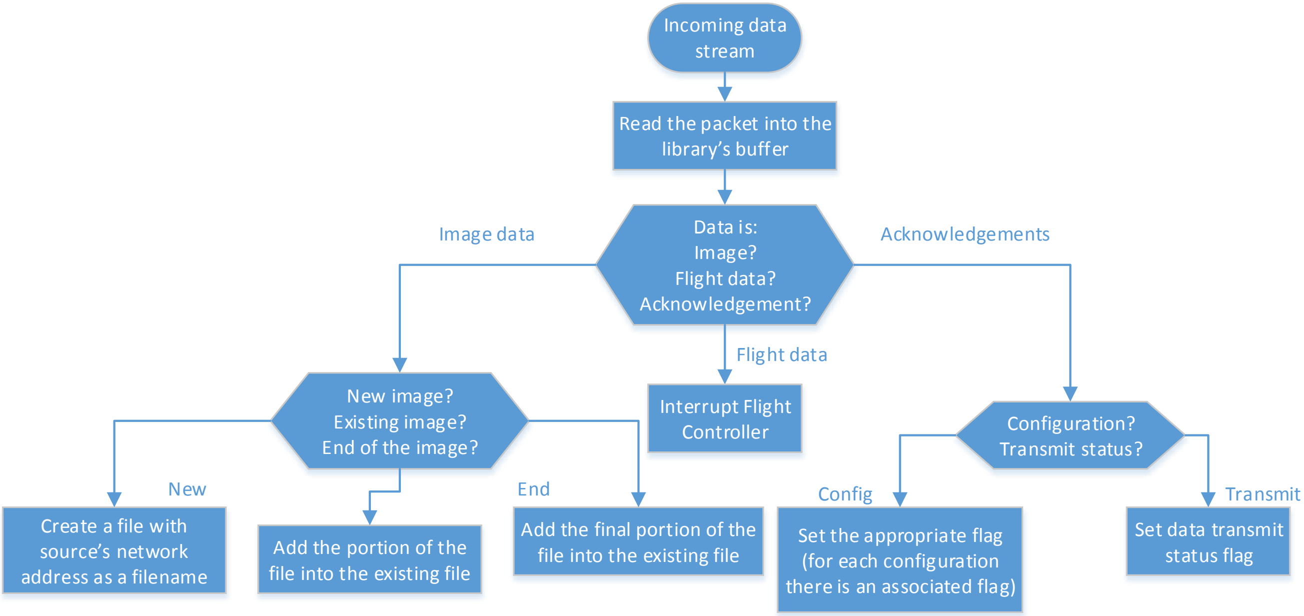

Fig. 4. Data processing

perform the same function calls to the API that the SPOT's flight controller would.

B. *Effective Data Throughput Measurements*

Our goal is to evaluate the effective throughout for a single and two-hop transmission. First, we measure the throughput when there is no movement of the nodes. Then we verify the same results with mobile scenarios. The effective throughput is calculated as follows:

$$\text{ET} = \frac{\text{Total number of bits (or bytes)}}{\text{Time it takes to send and receive a file (seconds)}} \tag{1}$$

The captured transmission duration includes the time it takes to acknowledge each packet. Without the acknowledgements the transmission becomes unreliable and as explained later, the transmitted photos may get corrupted at the receiver end. We are interested in the real (wall-clock) time it takes to transmit a given amount of data. The accuracy of the measurements and calculations are ± 1 second. For our application this is an acceptable accuracy because the environment around the dirigible does not change significantly within a second. All of the on-board sensors are sampled once every second.

1) One-hop Throughput: During the calculation of the throughput we take into account the packet header and the checksum bits, which constitute an additional 128 bits. The maximum payload per packet is 255 bytes, which is twice more than specified by 802.15.4 (maximum of 127 bytes). Our goal is to use the one-hop throughput as a reference for the two-hop transmission later, in order to find by how much the effective end-to-end throughput decreases in a multi-hop network using DigiMesh. If we assume that a one hop transmission takes y seconds, we expect that the worst case in a two hop transmission would take 2×y seconds.

TABLE I
ONE-HOP EFFECTIVE THROUGHPUT RESULTS

Image (bytes)	Total packets	Header (bytes)	Tx data (bytes)	Tx time (seconds)	Throughput (bps)
45397	179	2864	48262	11-12	32175-35100
112482	442	7072	119555	28-29	32981-34159

On average our communication framework achieves a throughput of 31017-33935 bps. The results are summarized in Table I. Comparing these results to the throughput of 40 Kbps, specified by the 802.15.4 standard [28], we conclude that the two values are close. The 802.15.4 standard specifies the throughput at the physical layer, while we were measuring the throughout at the application layer because this is the amount of time the framework takes to package, transmit, and reassemble the data.

2) Two-hop Throughput: In order to measure the effective throughput for a two hop transmission, we needed to make sure that all the packets are routed through an intermediate node. On average our communication framework achieves a two-hop throughput of 23328-24524 bps. The results are summarized in Table II. We observe that the transmission time increases compared to the one-hop case, resulting in end-to-end throughput decrease by 41-42%. AUGNet tests in [26] reported that with their implementation of DSR reactive

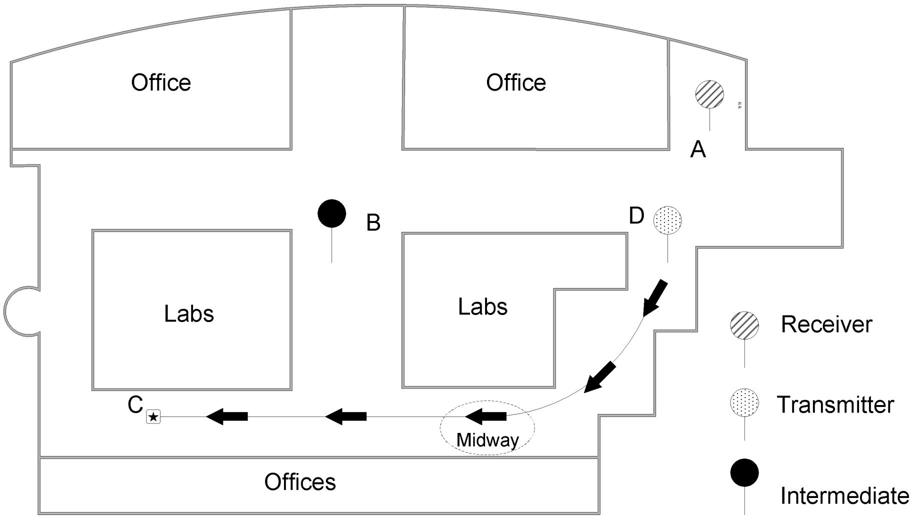

Fig. 7. Floor setup

TABLE II
TWO-HOP EFFECTIVE THROUGHPUT RESULTS

Image (bytes)	Total packets	Header (bytes)	Tx data (bytes)	Tx time (seconds)	Throughput (bps)
45397	179	2864	48262	15-17	22711-25740
112482	442	7072	119555	39-41	23328-24524

protocol, the effective throughput decrease by a factor of two to three (100-150%). Comparing these two results showed that the underlying routing in DigiMesh is faster than the IP based routing in AUGNet. The final point is that the transmitter power level does not affect the throughput. We performed the one-hop and two-hop tests with different power levels and found out that the throughput does not change as the transmit power levels change. As long as all the packets are received with RSS above the radio's sensitivity of -100 dBm, the throughput remains constant. This is expected because the 9XTend radio does not implement an adaptive modulation scheme, which would provide higher throughput at higher power levels and vice versa. The movement of the nodes also did not affect the throughput while the packets were received with an RSS above -100 dBm. This means that as long as the SPOTs are moving within the LOS communication range and maintaining the RSS above -100 dBm, the throughput will not change.

C. Reliability of The Mesh

The communication framework is being designed to operate on the mobile SPOT in an uncontrolled environment. Therefore, it is important to maintain a reliable connection between the radio nodes in the network for achieving the measured throughputs and receiving compressed photos without errors. The reliability of the wireless connections can be improved using software techniques and/or diversity techniques in the space domain. The latter is a future goal in the ongoing development. In this section we focus on the implemented software techniques with acknowledgements and checksum. We will show the benefit of using acknowledgements, at the cost of lower effective throughput, in order to control the serial flow of data, especially when transferring photos in the network. Acknowledgements become the only reliable way for controlling the SPOT in a no-LOS environment.

When sending photos, the designed API performs fragmentation, packetization, and reassembly of images. We are not using any additional image processing library and only work with the binary data. Our first design and development of the image fragmentation and packetization used only delays. The framework splits the JPEG encoded photos into packets and then assembles them back at the receiver. After sending each packet to the radio from the computer, the API pauses for 10-30 milliseconds. This delay was used to give the radio enough time for processing and sending the packet to the receiver. We assumed that the packets were arriving at the receiver in the same order as they were being sent by the transmitter. This assumption simplifies the reassembly of the photos. Examples of the received images are shown on Figure 8. As the results show this was not a reliable way of transmitting large files. Out of 10 transmissions of each photo, 5 to 7 of the received photos would be corrupted even if the link reliability is high.

The reason behind the corrupted images is that some of the packets were not received in the right order. Consequently the reassembly of the photos was erroneous. The images were transmitted over a link with a reliability of 98.87-99.04%. The reliability is computed as:

$$\frac{Size\ of\ received\ image}{Size\ of\ sent\ image} \tag{2}$$

Looking at the nature of the error it looks that the images were received correctly at the beginning but then some parts were flipped horizontally. Because our design relies on a

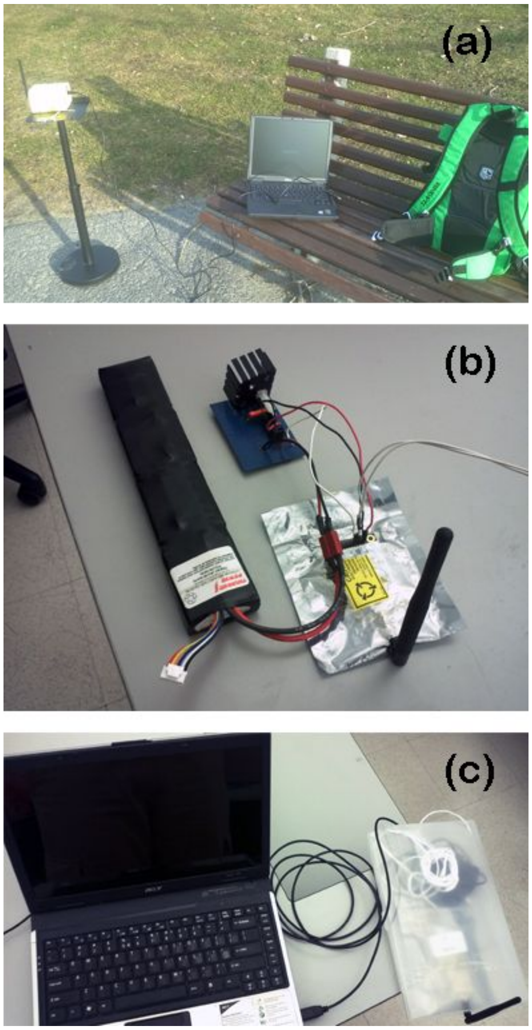

Fig. 6. Testbed setup

Fig. 8. Sent image (left) Received image (right)

correct sequence of packets this suggests that some of the packets were received in the wrong order while the 1-2% of total packets were lost. In a mobile wireless mesh it is possible that some packets can reach the receiver earlier than the others because the routes can change for each packet depending on the link's strength. For the next design revision we made use of the acknowledgements instead of delays to control the flow of data. Now, every time the packet is sent the API waits for the acknowledgement before sending the next one. This way it is guaranteed that all the packets will be received in the correct order before being assembled into the image. This solved the problem with the corrupted images and the receiver was able to reassemble correctly the image shown on the left of Figure 8.

D. Self-healing Links - Best Route Selection

Our goal is building a mesh that can rearrange its topology to maintain connections in a mobile radio network. In this section we perform a test scenario where an existing link breaks during the transmission of an image and a new route is chosen. In our test setup, we start the transmission of an image at point D of Figure 7 and move the transmitter as indicated by the arrows. As the transmitter gets close to the intermediate node at point B, the link between the transmitter and receiver becomes weak and eventually breaks before the transmitter can reach point C. When the link breaks the packets are routed through an intermediate node. During the transmission the API captures the RSS of each packet and stores it into a file. We then generate the plot shown on Figure 9, where each point indicates a single received packet. The graph on the Figure shows that as the transmitter is moved further away from the receiver the RSS drops. Up to about 45 meters one-hop transmission is taking place. After that there is a positive jump in the RSS values, indicating that the transmitter started to send the remaining of the image packets through a closer intermediate node. During the two-hop transmission and rerouting of the remaining packets no loss of data occurred because the images were received successfully. This result shows that combining our API with the radios, the framework maintains the shortest path during a continuous transmission as long as the received RSS is above the threshold of -100 dBm. When the transmitter fails to receive an acknowledgement it starts a new route finding procedure defined by the DigiMesh protocol. This demonstrates how DigiMesh dynamically selects the best route for a different node configuration.

E. Data Collision Handling

We are now going to analyse the performance of the mesh links in the presence of an interfering node. From Figure 7, we place the transmitter midway between the receiver and the intermediate node such that all three radios can communicate. The intermediate node is set to broadcast the same message every half a second, acting as an interference node. At the same time the transmitter sends an image to the receiver. The same scenario with more than three radios can be pictured on Figure 10.

We observe that the two separate data flows are interfering with each other at the shaded node. Our tests have shown that the shaded node slows down both data flows because the transmission time was multiplied by 4-6 times. The solution we developed to this case is assigning different network IDs to the transmitter and the receiver. This translated into a different hopping sequence since the radios are

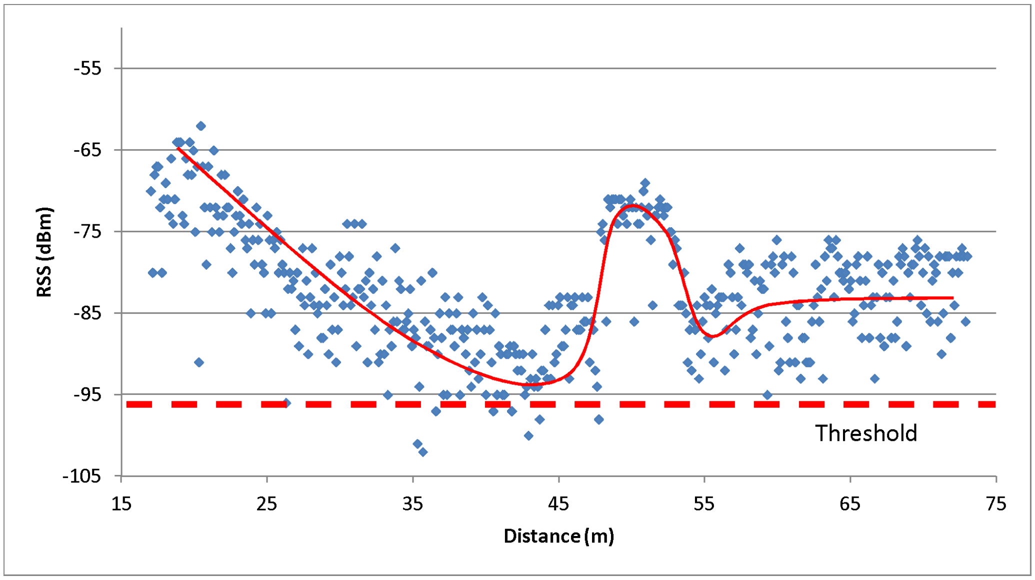

Fig. 9. Two hop RSS plot

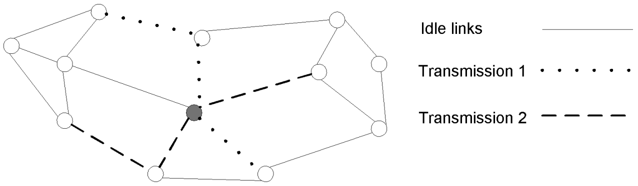

Fig. 10. Example of data flows in a mesh

communicating using frequency-hopping spread spectrum. In the DigiMesh network, only the radio nodes with the same network ID can exchange data (clusters). By default all radios are assigned `3332` network ID. We changed this to `1111` on the transmitter and the receiver node, which isolated the pair from the broadcasting intermediate node. Once the self-interference was eliminated, we achieved the same one-hop throughput discussed earlier. For the the case with multiple radios this result allows to prioritize data flow, by allowing higher priority data to flow over a shorter path as illustrated by a dotted line on Figure 11.

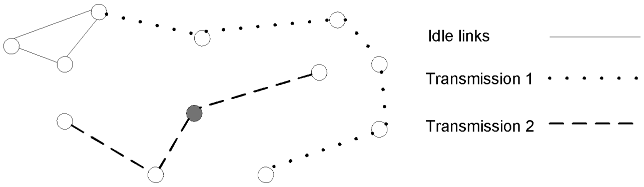

Fig. 11. Example of data flows in a mesh

F. Long-range Communication

Up to now we have demonstrated that the communication framework uses the DigiMesh routing protocol in order to successfully implement and maintain a transmission in a network with mobile nodes. The purpose of the following experiment is to demonstrate that a single hope within our framework has the capability of achieving 1+ km communication distance with 1-10 mW power and half-wave dipole antennas. The section over the Ottawa River between the Victoria Island and the Alexandra Bridge was chosen as the test site for our long-range experiment (Figure 12). These sites were chosen for two reasons: the line-of-site between the two locations and the elevation of 5-10 meters above the river on the bridge.

The tests were performed by transmitting images at different transmit power levels. The weakest RSS was captured to be -93 dBm when the transmit power of 1 mW was used. At 1 mW transmit power the transmission time was increased by 30% compared to the transmission at 1 W. After increasing the transmit power to 10 mW the same one-hop throughput results as earlier were achieved.

V. CONCLUSION

At the beginning of this research project we explained our main goal of building a long-range communication framework. Initially, our main application for the developed framework was integrating it into the fleet of autonomous search and rescue dirigible UAVs because it offers a longer endurance than most of the existing UAVs for the same price range of $1000-$5000. The design and development of the framework has been discussed along with the test results. In addition, our implemented framework can be used as a testbed to support research projects involving the performance evaluation of new emerging ad hoc routing techniques, distributed wireless sensor platforms, and multiple UAV control algorithms. As the main research problem stated, currently there is an urgent need for a decentralized mesh communication in several projects working on the development of cooperative multi-robot systems.

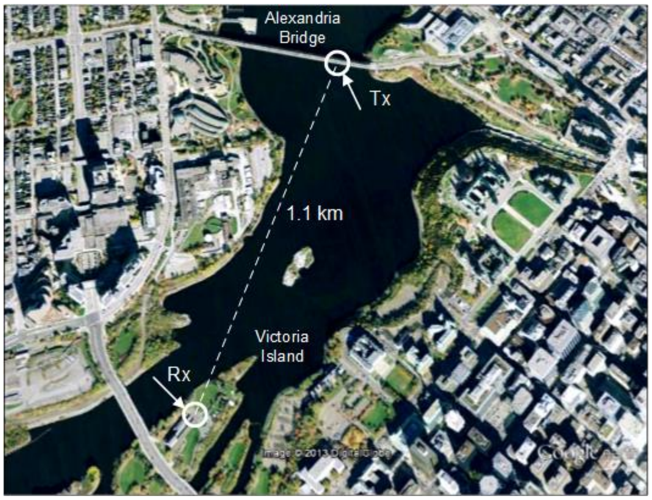

Fig. 12. Long-range test setup

ACKNOWLEDGMENT

The authors would like to thank The Natural Sciences and Engineering Research Council of Canada for funding the purchase of the radios and the required components for building communication and power interfaces.

REFERENCES

[1] S. Recoskie, A. Fahim, W. Gueaieb, and E. Lanteigne, *Experimental Testing of a Hybrid Power Plant for a Dirigible UAV*, Journal of Intelligent & Robotic Systems, 69:69-81, 2013.

[2] M. I. Lizarraga, David M. Ilstrup, G. H. Elkaim, and J. Davis, *Aerial Photography using a Nokia N95*, World Congress on Engineering and Computer Science, 2008

[3] S. Hrabar, T. Merz, and D. Frousheger, *Development of an Autonomous Helicopter for Aerial Powerline Inspections*, IEEE 1st International Conference on Applied Robotics for the Power Industry, 2010

[4] I. Maza, K. Kondak, M. Bernard, and A. Ollero, *Multi-UAV Cooperation and Control for Load Transportation and Deployment*, Journal of Intelligent and Robotic Systems, 57:417-449, 2010

[5] T. Laengle, T. C. Lueth, and U. Rembold, *A Distributed Control Architecture for Autonomous Robot Systems*, International Conference on Informatics and Control, 1997

[6] T. Lacroix, T. Alami, T. Lemaire, G. Hattenberger, and J. Gancet, *Decision Making in Multi-UAVs Systems: Architecture and Algorithms*, book title ”Multiple Heterogeneous Unmanned Aerial Vehicles”, 37:15-48, 2007

[7] J. Gancet, G. Hattenberger, R. Alami, and S. Lacroix, *Task Planning and Control for A Multi-UAV System: Architecture and Algorithms*, IEEE/RSJ International Conference on Intelligent Robots and Systems, 2005

[8] L. E. Parker, *Current State of the Art in Distributed Autonomous Mobile Robotics*, book title ”Distributed Autonomous Robotic Systems 4”, pp 3-12, 2010

[9] T. Balch and R. C. Arkin, *Communication in Reactive Multiagent Robotic Systems*, Journal of Autonomous Robots, 1:27-52, 1994

[10] D. Lucey, *Minicopters Take the Art of Photography to the Skies*, Unmanned Systems: Mission Critical magazine, 2:29-34, 2012

[11] Lehmann Aviation, *UAV Aerial-monitor Oil and Gas Wells, Facilities, Pipelines*, Unmanned Systems: Mission Critical Magazine, 58:11, 2012

[12] D. Lucey, *Virginia Tech Researchers Seek Out Mysterious Blankets of Biomaterials*, Unmanned Systems: Mission Critical magazine, 2:17-93, 2012

[13] T. H. Cox, C. J. Nagy, M. A. Skoog, I. A. Somers, and R. Warner, *Civil UAV Capability Assessment*, NASA Technical Report, 2004

[14] Z. Sarris, *Survey of UAV Applications in Civil Markets*, IEEE Mediterranean Conference on Control and Automation, 2001

[15] S. Rohde, N. Goddemeier, K. Daniel, and C. Wietfeld, *Link Quality Dependent Mobility Strategies for Distributed Aerial Sensor Networks*, IEEE Globecom Workshop on Wireless Networking for Unmanned Aerial Vehicles, 2010

[16] N. Goddemeier, S. Rohde, K. Daniel, and C. Wietfeld, *Communication Constrained Mobility and Topology Management for Relay Sensor Networks*, IEEE WCNC 2011 - Network, 2010

[17] G. Vachtsevanos, B. Ludington, J. Reimann, P. Antsaklis and K. Valavanis, *Modeling and Control of Unmanned Aerial Vehicles - Current Status and Future Directions*, Workshop on Modeling and Control of Complex Systems (MCCS), 2005

[18] S. Levy, *Aftershock: Unmanned Systems Help Japan Recover From Disaster*, Unmanned Systems: Mission Critical magazine, 1:18-22, 2011

[19] A. Ryan, M. Zennaro, A. Howell, R. Sengupta, and J. K. Hedrick, *An Overview of Emerging Results In Cooperative UAV Control*, 43rd IEEE Conference on Decision and Control, 2004

[20] M. J. Matczynski, *An Overview of Emerging Results In Cooperative UAV Control*, Master’s Thesis, Massachusetts Institue of Technology, 2006

[21] J. Chen, Q. Fei, and Q. Geng, *The Design of Data Link for Multi-UAVs*, 4th International Conference on Intelligent Human-Machine Systems and Cybernetics, 2012

[22] N. Goddemeier, K. Daniel, and C. Wietfeld,, *Coverage Evaluation Of Wireless Networks for Unmanned Aerial Systems*, IEEE GLOBECOM Workshops, 2010

[23] M. Wzorek, D. Landen, and P. Doherty, *GSM Technology as a Communication Media for an Autonomous Unmanned Aerial Vehicle*, 21th Bristol UAV Systems Conference, 2006

[24] A. Boukerche, *Algorithms and Protocols for Wireless, Mobile Ad Hoc Networks*, Wiley-IEEE Press, 2008

[25] E. Kohler, R. Morris, B. Chen, J. Jannotti, and M. F. Kaashoek, *The Click Modular Router*, ACM Transactions on Computer Systems, 2000

[26] T. X. Brown, B. Argrow, C. Dixon, S. Doshi, R. Thekkekunnel, and D. Henkel, *Ad Hoc UA V Ground Network (AUGNet)*, 3rd Unmanned Unlimited Technical Conference, 2004

[27] H. Wirtz, T. Heer, R. Backhaus, K. Wehrle, and D. Henkel, *Establishing Mobile Ad-Hoc Networks in 802.11 Infrastructure Mode*, 6th ACM international workshop on Wireless network testbeds, experimental evaluation and characterization, 2011

[28] Y. Xiao, *Handbook on Sensor Networks*, World Scientific Publishing Company, 2010