

IoT-Enabled Autonomous System Collaboration for Disaster-Area Management

Abenezer Girma, *Student Member, IEEE*, Niloofar Bahadori, Mrinmoy Sarkar, Tadewos G. Tadewos, Mohammad R. Behnia, M. Nabil Mahmoud, Ali Karimoddini, *Senior Member, IEEE*, and Abdollah Homaifar

Abstract—Timely investigating post-disaster situations to locate survivors and secure hazardous sources is critical, but also very challenging and risky. Despite first responders putting their lives at risk in saving others, human-physical limits cause delays in response time, resulting in fatality and property damage. In this paper, we proposed and implemented a framework intended for creating collaboration between heterogeneous unmanned vehicles and first responders to make search and rescue operations safer and faster. The framework consists of unmanned aerial vehicles (UAVs), unmanned ground vehicles (UGVs), a cloud-based remote control station (RCS). A light-weight message queuing telemetry transport (MQTT) based communication is adopted for facilitating collaboration between autonomous systems. To effectively work under unfavorable disaster conditions, antenna tracker is developed as a tool to extend network coverage to distant areas, and mobile charging points for the UAVs are also implemented. The proposed framework's performance is evaluated in terms of end-to-end delay and analyzed using architectural analysis and design language (AADL). Experimental measurements and simulation results show that the adopted communication protocol performs more efficiently than other conventional communication protocols, and the implemented UAV control mechanisms are functioning properly. Several scenarios are implemented to validate the overall effectiveness of the proposed framework and demonstrate possible use cases.

Index Terms—Architectural analysis and design language (AADL) and cloud computing, disaster area management, internet of things (IoT), message queuing telemetry transport (MQTT), unmanned aerial vehicle (UAV), unmanned ground vehicle (UGV).

I. INTRODUCTION

FIRST responders are required to operate immediately after a disaster strikes, such as earthquakes, floods, nuclear leakages, or explosions. In such situations, it is important to

quickly operate to save lives and reduce property damage. In order to do so, rescue teams have to face dangerous situations to rapidly gather the required information, to get a quick overview of the environment condition and to locate the victims. Besides that, timely identifying hazardous sources such as chemical leakage or extreme heat emission is very important to prevent further damages. Moreover, when disaster strikes the condition of surrounding environment dramatically changes, which makes it even more difficult for the responders to assess such an uncertain and complex environment.

Recently, efforts have been made in using teams of unmanned vehicles (UVs) such as unmanned ground vehicles (UGVs), to quickly and safely assess disaster areas [1]–[3]. The UVs collaboratively work with each other and with human operators to take multiple measurements to evaluate the environment. For instance, in the nuclear disaster that happened in Fukushima (Japan) in 2011, the deployment of UGVs equipped with sensors helped to quickly detect both hazard sources and victims [4].

Most existing studies on the deployment of a large-scale team of UVs have either remained as theoretical concepts or had only partially implemented [5], [6]. The existing UVs collaborative frameworks commonly lack the means of efficient and realistic communication protocols to enable effective wireless communication among resource-constrained devices in the framework [7]–[11]. To tackle this, some frameworks use conventional address-based communication protocols (such as HTTP) [10], [11]. However, studies have shown that such communication protocol is not optimized to be resource-efficient for mobile and resource-constrained devices [12], [13]. Besides, most of the existing frameworks can not extend network service to the autonomous systems, such as UAVs, working in network denied disaster environment and they do not also have a nearby UAVs' charging point. To address these issues, some existing frameworks [7]–[9], relied on the oversimplified assumption that a nearby functional network infrastructure and power source is always available for the UAVs to get network access and charge their battery, respectively. However, this assumption might be infeasible in disastrous or distant areas where the network coverage may not be available and charging point not easily reachable. More importantly, existing frameworks do not also apply a formal design verification method, which might hinder their practical feasibility given the complexity of the design and the real-

Manuscript received February 19, 2020; revised March 26, 2020, May 10, 2020; accepted May 28, 2020. This work was supported partially by Air Force Research Laboratory, the Office of the Secretary of Defense (OSD) (FA8750-15-2-0116), the National Science Foundation (NSF) (1832110), and the National Institute of Aerospace and Langley (C16-2B00-NCAT). Recommended by Associate Editor Giancarlo Fortino. (*Corresponding author: Abdollah Homaifar.*)

Citation: A. Girma, N. Bahadori, M. Sarkar, T. G. Tadewos, M. R. Behnia, M. N. Mahmoud, A. Karimoddini, and A. Homaifar, "IoT-enabled autonomous system collaboration for disaster-area management," *IEEE/CAA J. Autom. Sinica*, vol. 7, no. 5, pp. 1249–1262, Sept. 2020.

The authors are with the Department of Electrical and Computer Engineering, North Carolina A&T State University, Greensboro, NC 27411 USA (e-mail: aggirma@aggies.ncat.edu; nbahador@aggies.ncat.edu; msarkar@aggies.ncat.edu; tgdadewos@aggies.ncat.edu; mbehniap@ncat.edu; mnmahmoud@ncat.edu; akarimod@ncat.edu; homaifar@ncat.edu).

Color versions of one or more of the figures in this paper are available online at <http://ieeexplore.ieee.org>.

Digital Object Identifier 10.1109/JAS.2020.1003291

time limitation of the operation. These challenges and gaps between the theoretical research and implementation in the realm of the autonomous systems and wireless networks, limit the successful development and deployment of collaborative UAV networks in for disaster area operation.

This paper addresses the aforementioned gaps in practical realization of heterogeneous autonomous system collaboration for disaster area management tasks. We proposed a framework for enabling easy collaboration among UAVs, UGVs and first responders operating in a disaster area to make search and rescue operations safer and faster. A modern and efficient MQTT machine-to-machine (M2M) communication protocol is adopted to enable easy interactions among the framework components. Each UGVs and UAVs are equipped with multiple sensors and actuators to collect and share surrounding information with the first responders (remote operators) and execute a given command, respectively. While the UGVs assess the situation on the ground, the UAVs assist them in coordinating their activities by providing communication support and sensing the environment from the aerial view. A cloud-based remote control station (RCS) is used to enable first responders (human operator) to remotely monitor the operation from anywhere through the Internet.

Additionally, an antenna tracker is developed and used to extend the communication network coverage from a nearby functional infrastructure to the UAV operating in disaster or distant area. Beside, a clustering algorithm is implemented in the framework to maximize the network coverage provided by the UAVs to the UGVs. Further, a control mechanism that enables a UAV to land on a moving UGV is implemented for application such as recharging UAVs and transport them to a distant location when required. Next, the proposed architecture is analyzed and evaluated using architectural analysis and design language (AADL) tool.

The main contributions of this paper are summarized as follows:

- 1) A collaborative heterogeneous autonomous system framework is designed and implemented for disaster area management that leverage the combined benefits of UAVs, UGVs, IoT technology, cloud connectivity, and antenna tracker.

- 2) An antenna tracker is developed to extend network coverage to the UAVs and an effective UAV resource allocation mechanism is implemented using a clustering algorithm technique for maximizing the UAVs' network coverage area.

- 3) A control mechanism for landing UAV on moving UGV is implemented for recharging UAVs as well as to transport them to long distance.

- 4) AADL is used to analyze and evaluate the end-to-end delay of the proposed framework. AADL is used in the automotive and aerospace industry for years to model and analyze complex systems for their performance [14]–[16], but to the best of our knowledge, this is the first work that uses AADL for timing analysis in an IoT-enabled autonomous system.

Moreover, we conducted various practical testing scenarios to evaluate the overall performance of the proposed system. The implementation and demonstration videos are available

online¹.

The remainder of this paper is organized as follows: Section II presents reviews of related work. Section III presents the system model for the proposed framework. The system components development is discussed in Section IV. Section V discusses the AADL analysis. Finally, the results are discussed in Section VI, and the conclusion is presented in Section VII.

II. RELATED WORK

Extensive research has been performed in the areas of autonomous systems and wireless networks, however, the intersection, WASN is fairly unexplored research area. Recently, utilizing systems of autonomous agents have attracted much attention to assist human operators in performing difficult and dangerous missions, such as search and rescue in disastrous environments and exploration of unknown environments [17]–[20]. In most of these works, the Ad-hoc communication network has been utilized to connect the autonomous agents. However, an Ad-hoc network cannot be efficient especially in a post-disaster areas as changes in the environment caused by the disaster, hinders reliable communication among agents. For example, blockages may delay the agents discovery process or deteriorate communication link quality [21]. Therefore, a central node (such as base station) is required to provide communication network coverage, relay the communication among agents, and monitor the environment to improve agents' performance.

UAVs are envisioned to be an important technology for WASN implementation due to their numerous advantages, such as swift and flexible deployment and providing line-of-sight (LoS) communication channels. Most of the existing research investigating UAV communications mainly focus on two aspects: connectivity and information collection.

UAVs for Network Connectivity: UAVs are used in the literature to improve the network connectivity through integrating onboard communication devices [22]–[24]. In these schemes UAVs act as relaying points between autonomous agents [25], [26] to extend the infrastructure coverage in distant/post-disaster areas, where communication network coverage is not available. Similarly, in [22], [24] network of UAVs has been utilized to maintain the communication among IoT sensor nodes. In [27], SkyLiTE system is designed to provide LTE connectivity for users by extending the network coverage. Other areas have utilized UAVs for network coverage such as public safety [9], [28], vehicular [29]–[32], and emergency [33], [34] applications. In addition, various mechanisms were proposed to improve the performance of UAV-assisted communication networks [35]–[39].

UAVs for Information Collection: UAVs have been used in the literature to collect information from the environment autonomously through onboard sensing devices such as sensors and cameras [40]–[45]. The collected information is either processed by the UAV and used to improve the performance of the system or sent to a remote control center

¹ <https://github.com/Abeni18/IoT-Enabled-Autonomous-UAV-UGV>

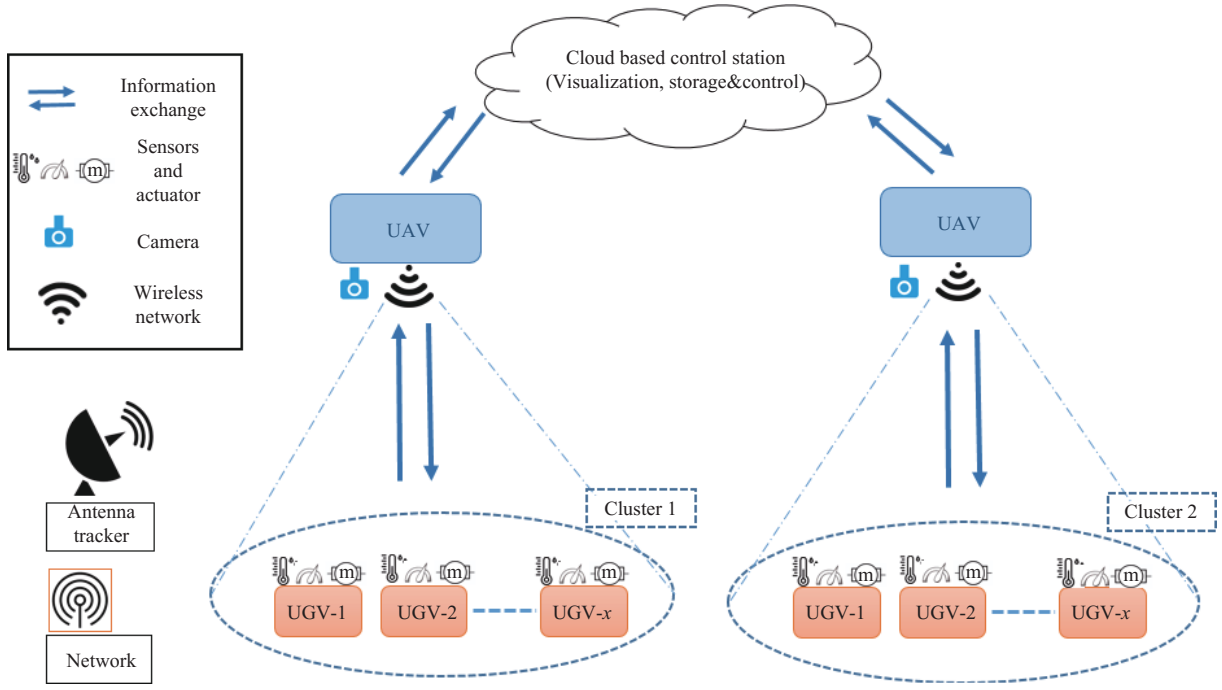


Fig. 1. System architecture of IoT enabled autonomous system.

for further analysis. For instance, in [40] UAVs have been used to monitor traffic conditions and pedestrians for e-health applications. A network of UAVs is used in [41] to provide Internet coverage and facilitate routing among ground vehicles by collecting network and road information. Moreover, a network of camera-equipped UAVs have been used in various applications, such as monitoring forest health [42], hazards and disaster risk monitoring [43], and civil infrastructure monitoring [44].

Most of the aforementioned studies are theoretical concepts, while a practical low-cost implementation of a platform that combines these functionalities of UAVs along with UGVs through an efficient M2M communication protocol, is lacking in the literature. To address these gaps, we propose a heterogeneous framework in which UGVs, UAVs and human operators at the remote control station work collaboratively to accomplish a mission. In this platform, UGVs collect environmental information through their onboard sensors. UAVs monitor UGVs and the search environment from above, while relaying the UGVs' collected information to the remote control station. An Antenna tracker is designed and developed which connects to the UAVs to extend the infrastructure coverage to UGVs and enable UGVs to explore farther locations. Finally, the performance of the proposed framework is verified through AADL to ensure that the design meets predefined system requirements.

III. SYSTEM MODEL

The proposed WASN framework is composed of UGVs, UAVs, and RCS as shown in Fig. 1. This framework is proposed to be deployed in a disaster-affected areas to assist first responders to locate the victims and hazard sources by observing the environment. Considering the high likelihood of communication infrastructure malfunctioning in these areas, the use of UAV in combination with antenna tracker is

proposed. While the UAV provides the communication support in the proposed framework, the antenna tracker extends the communication coverage from a nearby infrastructure to the UAVs by tracking their location. The UGVs autonomously navigate in the environment searching for the given targets, while collecting and forwarding information to RCS with the assistance of the UAVs. The UAVs perform surveillance and monitoring activities through onboard cameras from the above. As shown in Fig. 1, to use the available UAV resources efficiently, the UGVs get clustered into smaller groups, each corresponding to an available UAV in the framework. Then the UAV follows the UGVs by hovering at the center of the group for providing maximum network coverage. We also consider the case that UGVs and UAVs have to continuously operate for an extended period or travel to a long distance destination which can drain the UAVs' batteries faster. Therefore, UAVs might need to recharge their battery multiple times during a mission. In such cases, UGVs can serve as recharging units for the UAVs or can carry UAVs to a farther location to save the battery of the UAVs. Further, in the cases when UAVs cannot fly in certain areas, due to unfavorable geographical/environmental conditions or limited flight permissions, UGVs can transport the UAVs to the target destination of operation. To address these, we implemented a control technique that enables UAV to land on moving UGV target.

IV. SYSTEM COMPONENTS DEVELOPMENT

In this section, the materials, software, and processes used for the hardware development of the system components are provided. As shown in Fig. 1, the system components are UAVs, UGVs, RCS, antenna tracker, and IoT modules.

A. Design and Development of the UGVs

The UGVs are required to autonomously move on the ground to collect important sensor information and execute

commands sent from the RCS. Two types of UGVs have been integrated into this work, in-lab built UGVs and Jackal commercial UGVs. As shown in Fig. 2, the structural design of the in-lab built UGVs is made up of four-wheel Rc-car chassis that is controlled by Wemos microcontroller and Adafruit motor drive. We integrate an ultrasonic sensor in front of the vehicle for enabling the obstacle avoidance system for autonomous navigation purposes. Ten UGVs are developed for testing different collaboration scenarios.

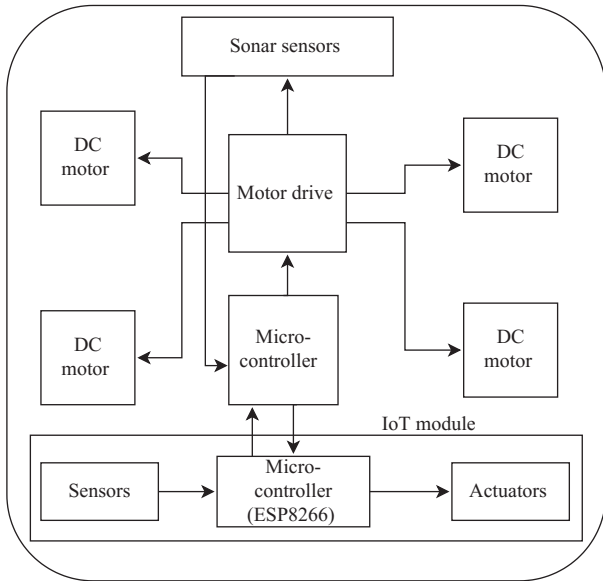


Fig. 2. In-lab built Rc-UGV structural design.

Moreover, a commercial weatherproof Jackal² UGV has been integrated into our platform. The main reason for utilizing Jackal UGVs is their suitability for outdoor harsh environment. Another reason is to demonstrate the integration capability of the developed framework with different types of UGVs. As described in [46], the Jackal UGV is an autonomous mobile outdoor robot introduced by Clearpath Robotics with robot operating system (ROS) full integration for autonomous capability.

The external dimensions of the UGV is 508 mm × 430 mm × 250 mm and the weight is 17 kg. It can carry a maximum payload of 20 kg with a maximum speed of 2 m/s. It has a Intel core i5 Dual-core, 2.9 GHz processor, 4 GB of RAM and 128 GB of SSD. The onboard sensors are inertial measurement unit (IMU) and wheel encoder. Different sensors such as LIDAR, camera, sonar can be integrated with it easily. The navigation system has been developed using the ROS as a middleware platform. During the experiment, both Jackal and in-lab built UGVs navigate in the given environment while avoiding collision with obstacles and other UGVs.

B. Design and Development of the UAVs

The UAVs are required to provide maximum communication support for the UGVs and sense the environment from aerial view. The UAVs have been developed in-lab using a quadcopter architecture with the mathematical model

described in [47], [48]. The design procedure described in [49], [50] has been followed to fabricate the UAVs. Development of UAVs from scratch and using open-source software ensure complete control over the UAV from its hardware to the software. It is not always possible to access all the components of the commercial UAVs, as only some limited control interfaces are provided. Also, the developed UAV can be customized in the sense that new sensors and features can be added easily. As shown in Fig. 3 the UAVs are built using off-the-shelf components available in the market. An iFlight Carbon Fiber Frame Kit has been used as an X-shape configuration frame. In addition, four SunnySky X2216 Outrunner Brushless motors have been used as the actuators of the UAV. Besides, Nidici 40A electronic speed control (ESC) is used for controlling the speed of the motor. Pixhawk Flight Controller has been used for low-level position and speed control of the UAV. For high-level data processing and planning, we used Raspberry-Pi 3B+. After the UAV has been developed, in order to design the UAV controller, the model parameters are measured as listed in Table I.

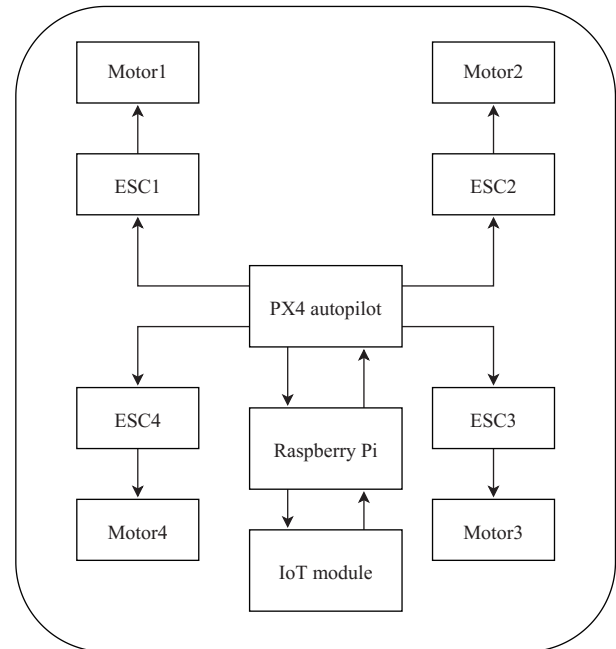


Fig. 3. UAV structural design.

The performance of the developed UAV has been tested under different scenarios. The UAV was tested successfully with a 250 gram external load, during which it flew stably. The offboard control mode of PX4 firmware enables UAVs to be sent to a 3D target location by initiating command from an external computer. PX4 autopilot uses a PID controller to control the position of the UAV. ROS in Raspberry-Pi has been used to provide the target location. VICON³ motion capture system has been used to provide the position of the UAVs in the lab environment [51]–[53]. The UAVs have been developed using publicly available libraries such as PX4 firmware and ROS.

² <https://clearpathrobotics.com/jackal-small-unmanned-ground-vehicle/>

³ <https://www.vicon.com>

TABLE I
UAV's MODEL IDENTIFICATION PARAMETERS

Parameter	Description	Measurement
m	Mass of quadcopter	1.095 kg
d	The distance between effective point of torques and the UAV's center of gravity	0.15 m
g	Gravity acceleration	9.80665 m/s ²
C_T	Thrust coefficient	0.0032
C_Q	Torque coefficient	0.028
k_m	Motor coefficients	38
b		87.23
j_{xx}	Moment of inertia	0.0025684
j_{yy}		0.0031458
j_{zz}		0.0052697

The performance of the developed drone has been tested in [54], while carrying out a search operation. In this work, an ambiguous scenario has been developed and UAVs' behavior has been predicted using a data-driven technique. For more information, the reader may look into [54].

C. Design and Development of the RCS

Similar works in the literature have suggested building and installing the control station close to the operation area [9], [11], however, using a cloud-based RCS reduces the cost and makes the control center to be easily accessible by the human operator and the UAVs. Cloud computing is a flexible, and economical approach for implementing real-time information exchange between users and wireless devices at any time with extensive coverage and quality. Researchers have investigated theoretical ways to connect dynamic IoT enabled autonomous systems to the cloud [39], [55], however, this work is mainly focuses on the actual deployment and testing of such systems.

IBM Bluemix cloud service is one of the IoT clouds that is widely used by industries, researchers and practitioners⁴. MQTT communication protocol was also developed by IBM with easy integration to work with the IBM cloud services [56]. Accordingly, the IBM Bluemix cloud service is used for developing the RCS in this work. There are three main purposes of the control station as shown in Fig. 1 and also discussed below:

1) *Collecting Sensor Data*: UAVs collect the sensor data generated by UGVs and then forwarded them to the RCS, where the data is stored in the database for further analysis.

2) *Real-time Visualization*: Visualization presents the data in an easily digestible format for the human operators who monitor the UGVs' and UAVs' activity.

3) *Remote Control*: As the UGVs and UAVs are operating remotely, the human operator can access the cloud through the Internet from anywhere and use it to send commands and control the framework's operation.

D. Design and Development of the Antenna Tracker

In order to extend the flying range of UAVs that are providing network coverage to the UGVs by connecting to nearby network infrastructure (such as nearby base-stations),

an antenna tracker system has been designed and developed, as shown in Fig. 4, which further extends the network coverage from the network infrastructure to the UAVs using its high-gain directional antennas.

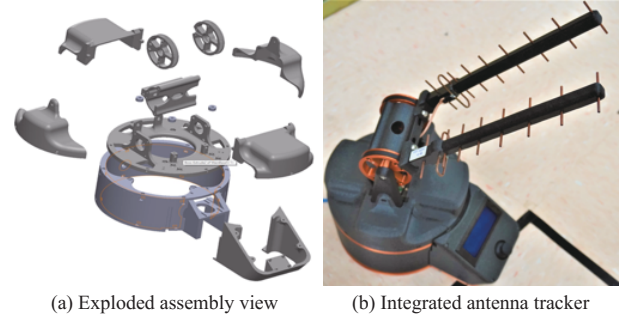


Fig. 4. Antenna tracker design and integration.

The antenna tracker is equipped with a control system, which receives the geographical coordinates of the UAV and aligns its antenna towards the obtained coordinates. Specifically, the antenna tracker is equipped with the two 2.4 GHz Yagi antennas with opposing circular polarization with a measured gain of 12.5 dBi capable of 2×2 MU-MIMO communication. As shown in Fig. 5, the antenna working frequency is measured using spectrum analyzer, in which the measurement results show that the developed antenna is precisely working in WiFi range (2.4 GHz).

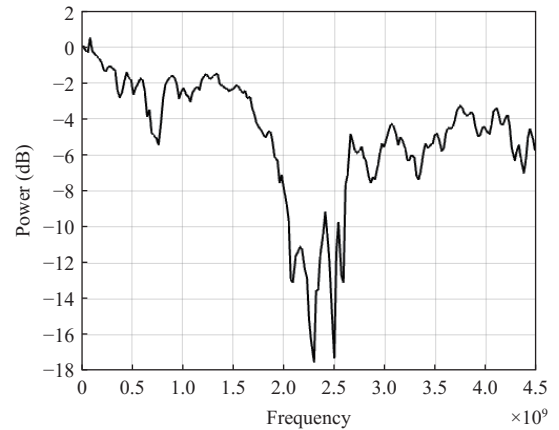


Fig. 5. The designed antenna tracker working frequency as measured by the network analyzer.

Seamless operation of the proposed framework requires a robust communication network that can facilitate interactions among the network components. In the next subsection, we discuss the communication protocol for M2M communication among UGV-to-UGV, UGV-to-RCS, and UAV-to-RCS.

E. Machine-to-Machine Communication

The use of a conventional addresses based communication protocols, like HTTP, in disaster area might be very problematic [12], [13], since the operation area during disaster is usually uncertain and highly dynamic as any or all of the devices may fail or change their network address at any time. Fortunately, this problem can be addressed by using data-

⁴ <https://www.ibm.com/cloud>

centric communication protocols which allow the devices to communicate based on their contents and interests rather than their network addresses. MQTT is widely used as communication protocol in the literature. MQTT protocol follows the publish/subscribe (pub/sub) technique for enabling a dynamic system topology [57], [58]. This protocol decouples the senders and receivers which makes it suitable for uncertain networks. MQTT enables low-power, low computational and bandwidth-constrained nodes to communicate with several devices without bearing the full communication burden [56], which perfectly matches the system requirements for the disaster area operations. Therefore, MQTT is used as the communication protocol to enable communication among the network components.

In MQTT, entities that generate information are called *publishers* and those who receive information are called *subscribers*. Both *publishers* and *subscribers* are referred to as MQTT clients. Since, the UGVs, UAVs, and RCS generate and receive messages, they are all publishers and subscribers. The central communication relay collects information from the publishers and then filters and forwards it to the subscribers through what is referred to as the *broker*.

MQTT offers three types of end-to-end quality-of-service (QoS) [56]. We have used the top two highest QoS', namely, QoS-1 and QoS-2 in this work. As QoS-2 guarantees one-time message delivery, it is implemented for sending control signals to the UGVs from RCS. Whereas QoS-1 guarantees message delivery with a shorter time than QoS-2, it is used for publishing sensory data to the broker. QoS-0 is not used because it does not guarantee the message delivery.

For scalability and easy integration, the hardware used to implement MQTT is developed as a separate module, referred to as the IoT-module. The UGV and the UAV IoT-modules are developed to handle the MQTT based M2M communications among the UGVs, UAVs, and RCS.

UGV IoT Module: As shown in Fig. 6, the UGV IoT-modules are developed from an off-the-shelf, low-cost WiFi-enabled microcontroller ESP8266 (NodeMCU). The NodeMCU is a low-power board that performs its functions with a full transmission control protocol TCP/IP stack and other peripheral devices essential for data processing [59]. The IoT module has been programmed in C++ and Arduino integrated development environment (IDE) to collect data from the sensors that are connected to its general purpose input-output (GPIO) ports. It then forwards them to the *broker* on the UAV. The IoT modules have a serial communication port (Tx and Rx) that enables serial communication with a variety of microcontrollers/embedded computers that exist on different types of UGVs (as discussed in Section IV-F).

UAV IoT Module: The UAV IoT module is built from Raspberry-pi 3 B+ model to serve as the MQTT *broker*. With 1 GB RAM and a quad-core ARMv7 processor, low-energy consumption and WiFi capability, the Raspberry-Pi is a powerful and economical computational device to be used as broker in our network. An open-source Eclipse mosquito⁵ message broker software is used to program the Raspberry-Pi to serve as MQTT broker. Besides, UAVs' IoT module is also

responsible for collecting important sensor information to be used for surveillance purposes.

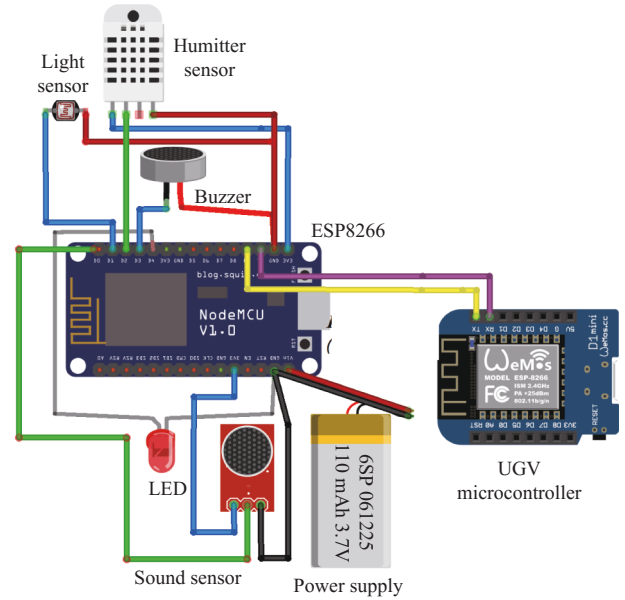


Fig. 6. UGV's IoT module design.

F. System Integration and Operation

1) UAVs and UGVs Integration: As shown in Fig. 7, the serial port is used to integrate the IoT module with the UGVs' onboard computer/microcontroller. The serial port of the UGV IoT module makes it versatile to be integrated with different types of UGVs. The IoT module is integrated with the two types of UGVs used in our system, in-lab built (Rc-UGV) and market acquired (Jackal UGV). The serial port of the IoT module is directly connected to the Tx and Rx port of the microcontroller of the Rc-UGV. For the Jackal UGV, we used a USB-to-Serial converter to connect the IoT module to the onboard computer of the Jackal UGV.

The UAV-IoT module is integrated with the UAVs' onboard computer through GPIO ports to serve as the network broker as well as to communicate with the UAV when necessary. For instance, UAV receives commands, such as takeoff and landing, from RCS through the IoT module. In addition, the UAV-IoT module is connected with a camera to monitor the UGVs and the surrounding environment.

2) System Operation: In order to maximize the network coverage during the UAVs operation, a K-means clustering algorithm is used to cluster UGVs and find the optimal UAV hovering position. Each UAV is located at the center of the UGVs cluster, therefore, K in the K-means clustering algorithm is equal to the number of UAVs. In order to maintain a safe flight altitude, each UAV flies at a pre-fixed different altitude. A PID controller is implemented to enable the UAVs to keep track of the center of clusters. The clustering algorithm updates the center of the cluster every $t_{\text{update}} = 3$ s, then UAVs adjust their positions accordingly. A 3-second update time has been chosen experimentally based on the time required by the UAV to run the clustering

⁵ <https://mosquitto.org/>

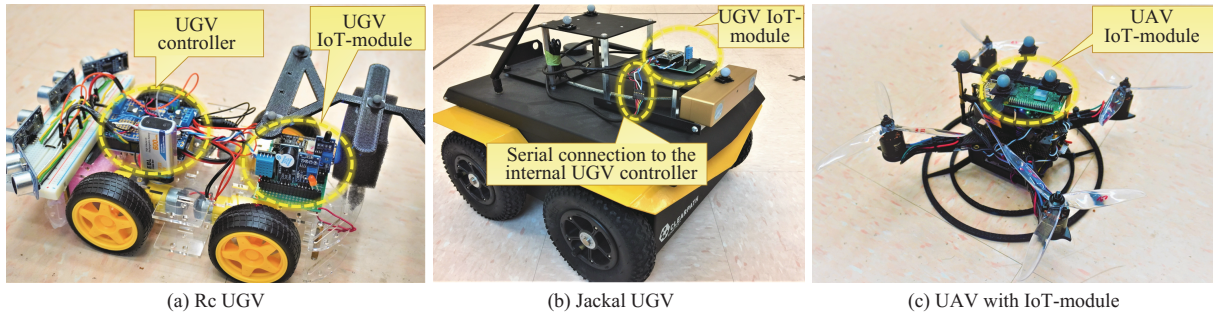


Fig. 7. IoT module integration with UGVs and UAV.

algorithm and adjust itself to the new cluster center. A block diagram of the whole process is shown in Fig. 8.

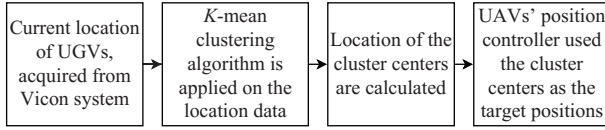


Fig. 8. The block diagram of the navigation process of the UAVs.

V. SYSTEM DESIGN VERIFICATION

AADL formal verification technique has been adopted as part of the system design process to verify the performance of the proposed system framework and to avoid the late detection of error [60]. The naive approach for system timing analysis is to simply add the latency contribution of each system's component. However, such basic analysis technique does not support complicated scheduling and architectural artifacts. In this work, AADL verification has been used in an iterative manner starting from system design stage, to verify and suggest modification of the developed prototypes, to the development of the final components discussed in Section IV. We have implemented AADL using the OSATE⁶ modeling framework to perform timing analysis.

The overall system end-to-end (UAV-UGV) latency should be satisfactory in order to send and receive the information in a timely manner. We have assumed that latency in the system could occur due to processing delay, sensing delay, scheduling delay, and transfer-time delay. In order to analyze these types of latency, AADL was used to define three constructs: 1) **source**: delay originates from a component; 2) **sink**: delay terminates within a component; 3) **path**: delay path from an incoming port to an outgoing port. Each module (UGV and UAV) is decomposed into its sub-components that provide a high fidelity model enabling a more reasonable estimation of the latency in that module. A delay as an interval has been assigned between tasks that could be scheduled periodically or asynchronously.

The framework is modeled using the building blocks of AADL, which are components, connections, and properties. For instance, as shown in Fig. 9, the UGV is modeled according to the AADL building blocks, where sensors, processors, and actuators are the components. The interaction between Wemos-D1 (central controller) and sonar sensors and

the interaction between Wemos-D1 and Adafruit motor driver are modeled as connections. Simultaneously, the interaction between the communication unit (ESP8266 microcontroller) with a sensor and its connection with Wemos-D1 are also modeled as connections. Similarly, AADL artifacts are used to model the UAV as shown in Fig. 10.

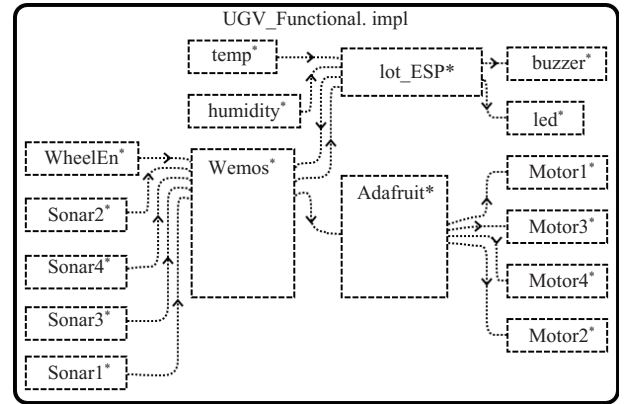


Fig. 9. AADL model of the UGV.

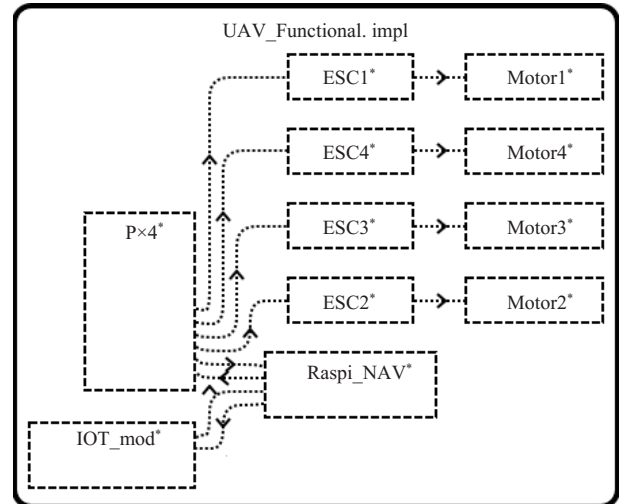


Fig. 10. AADL model of the UAV.

Once each subsystem has been modeled and assigned with the appropriate parameters, the AADL timing analysis tool is used to automatically compute the end-to-end delay of the system. Based on the result of the timing analysis, if a violation is reported, the system designer modifies the

⁶ <https://osate.org/>

structure, hardware, connections or the scheduling algorithms of the system to satisfy the timing requirement.

VI. RESULTS AND EVALUATIONS

In this section, we elaborate on the implementation of the proposed system model defined in Section III. The developed system is evaluated from three perspectives: AADL formal verification, control system, and communication system. The following subsections describe the scenarios used in our evaluation.

A. AADL Formal Verification

In order to compute the end-to-end latency of a system, the estimated delay that each component contributes to the total delay of the system must be provided as an input. This information can be obtained either from experimental measurements or by referring to the data-sheets of each component. For example, to estimate the UAV-UGV communication time, the transmission delay and hardware delay are required to be considered. Since the communication is performed through the IoT modules on UAVs and UGVs, the delay analysis can be estimated as follows:

1) *Hardware Delay*: The delay can be computed by considering the computational power of the processing units and the adopted type of software used to process the inputs. The latency introduced by the processing units and the sensors are obtained from their data-sheets [61]–[64]. However, as the processing time is not easily commutable, we estimate the processing time considering the speed of the CPUs.

2) *Transmission Delay*: The transmission latency is the time required to send a signal from UAV (UGV) transmitter to UGV (UAV) receiver through the air interface using 2.4 GHz frequency range [65].

The OSATE AADL tool is used to calculate the end-to-end UAV-UGV delay using the above information, as shown in Table II.

TABLE II
SYSTEM LATENCY ANALYSIS

Path	WC	TC	BC
Transmission delay	215 ms	184 ms	160 ms
UGV hardware delay	110 ms	98 ms	88 ms
UAV hardware delay	166 ms	136 ms	120 ms
Total end-to-end	491 ms	418 ms	395 ms

AADL provides the worst case (WC), typical case (TC), and best case (BC) end-to-end latency values of the overall testbed. As shown in Table II, for the WC, a message sent from UGV is received and processed by the UAV within 491 ms. In case that the UAV-to-UGV latency is more than 491 ms, the system violates the timing requirement and will need to be modified. The modification could include changing or improving the processors, sensors, or actuators. In an extreme case, a complete architectural redesign may be required. For example, timing analysis performed on an older design version of the UAV showed that all the timing

requirement cannot be fulfilled if only a single microcontroller handles all the UAV's operations. Therefore, the UAV system architecture was redesigned such that the UAV's tasks are broken down and executed in parallel on separate microcontrollers. Accordingly, as shown in Fig. 10 and discussed in Section IV-B, in the current version of the UAV system design, contains three independent but interconnected microcontrollers that handle the following task separately:

- 1) *IoT module*: handles the communication.
- 2) *PX4 autopilot*: controls the UAV flight system.
- 3) *Raspberry-pi*: runs the clustering algorithm.

B. Communication Performance Analysis

Wireshark packet tracer⁷ is used to record the network traffic and conduct the performance analysis. The following four metrics are used to assess the communication performance of our system.

- 1) *Response Time*: the time required to establish the communication link.
- 2) *Data Transfer Rate*: the average number of bits per unit time (bits/s) passing through a communication link during the publishing and subscribing process.
- 3) *Packet Size*: the average data packet size used in a communication between two nodes.
- 4) *Antenna Range*: how far an antenna can transmit a signal.

We began by evaluating the system performance through formally analyzing and formulating the time required to exchange messages between the publisher and subscriber, which is denoted in this paper by τ . The *publishers*, *subscribers*, and *broker* use MQTT protocol over TCP with standard QoS-1. The delay between the publisher and subscriber, (τ), can be decomposed into two parts: publisher-to-broker delay, τ_{Pub} , and broker-to-subscriber delay τ_{Sub} , and can be written as

$$\begin{aligned} \tau &= \tau_{\text{Sub}} + \tau_{\text{Pub}} \\ \tau_{\text{Sub}} &= \tau_{\text{ConReq}}^{\text{PB}} + \tau_{\text{ConReqACK}}^{\text{BP}} + \tau_{\text{Con}}^{\text{BP}} + \tau_{\text{ConACK}}^{\text{PB}} \\ &\quad + \tau_{\text{Pub}}^{\text{PB}} + \tau_{\text{PubACK}}^{\text{BP}} + \tau_{\text{d}}^{\text{sub}} \end{aligned} \quad (1)$$

$$\begin{aligned} \tau_{\text{Pub}} &= \tau_{\text{Con}}^{\text{SB}} + \tau_{\text{ConACK}}^{\text{BS}} + \tau_{\text{Sub}}^{\text{SB}} + \tau_{\text{SubACK}}^{\text{BS}} \\ &\quad + \tau_{\text{Pub}}^{\text{BS}} + \tau_{\text{PubACK}}^{\text{SB}} + \tau_{\text{d}}^{\text{pub}} \end{aligned} \quad (2)$$

where τ_x^y represents the delay of action x . The details of the parameters subscription can be found in Table III. As shown in (3), it has been assumed that the time for sending messages from the publisher to the broker and sending messages from broker to subscriber are symmetrical, i.e.,

$$\begin{aligned} \tau_{\text{Con}} &= \tau_{\text{ConReq}}^{\text{PB}} = \tau_{\text{Con}}^{\text{PB}} = \tau_{\text{Con}}^{\text{SB}} = \tau_{\text{Sub}}^{\text{SB}} \\ \tau_{\text{Pub}} &= \tau_{\text{Pub}}^{\text{PB}} = \tau_{\text{Pub}}^{\text{BS}} \\ \tau_{\text{ACK}} &= \tau_{\text{ConReqACK}}^{\text{BP}} = \tau_{\text{ConACK}}^{\text{BP}} = \tau_{\text{PubACK}}^{\text{BP}} = \tau_{\text{ConACK}}^{\text{BS}} \\ \tau_{\text{SubACK}}^{\text{BS}} &= \tau_{\text{PubACK}}^{\text{SB}}. \end{aligned} \quad (3)$$

Accordingly, the delay in (2) can be written as

$$\tau_{\text{Pub}} = \tau_{\text{Sub}} = 2\tau_{\text{Con}} + \tau_{\text{Pub}} + 3\tau_{\text{ACK}}. \quad (4)$$

It is worth noting that case QoS-2 is used, additional

⁷ <https://www.wireshark.org/>

TABLE III
DESCRIPTION OF THE PARAMETERS USED IN (2)

Parameter	Description
P, B, S	Publisher, Broker, Subscriber
ConReq, Con,	Connection request, Connection
Pub, Sub	Publish, Subscribe
ACK.	Acknowledgement
ConReqACK, ConACK	Connection request ACK, Connection ACK.
PubACK, SubACK	Publish ACK, Subscribe ACK.

acknowledgment time should also be considered in (4), to account for the extra step taken in QoS-2 for guaranteeing single message delivery among nodes.

The average time required to establish a connection (τ_{Con}) and publish a message (τ_{Pub}) are examined using Wireshark packet tracer. Equation (1) is used to calculate the τ_{Pub} for QoS-1 is 2.879 s. This result is verified to assure that the system delay satisfies the AADL timing requirement. As discussed in the previous section, the AADL timing analysis requires a single one-way end-to-end delay to be less than 0.491 s. Using (4) and end-to-end delay in Table II, total time required for publishing/receiving a full message have to be less than 2.946 s in order to satisfy the AADL-WC requirement. The measured time for publishing/receiving a message is 2.879 s, which is less than the AADL-WC required time. This indicates that the system sufficiently satisfies the AADL timing requirement. This also verifies that the system components are working efficiently as expected.

Next, we examined the impact of the number of UGVs communicating with the UAV on the achievable data-transfer-rate. We increased the number of UGVs communicating simultaneously to a UAV from one to four, while recording the network traffic. Each experiment (1 UAV communicating to 1, 2, 3 or 4 UGVs) was conducted for time span of 360.24 s with multiple repetition. The recorded network traffic during the experiment was analyzed using Wireshark, and the average data-transfer-rate is then calculated. As shown in Fig. 11, when the number of UGVs concurrently communicating to the UAV increases, the average data-transfer rate proportionally decreases. This shows that overloading the UAV by simultaneously connecting it to multiple UGVs must be avoided in order to maintain high data-transfer-rate. As an alternative, if the number of UGVs have to be increased, the number of UAVs also has to be increased proportionally to provide the required communication support. In our work, we have used two UAVs for eight UGVs, where each UAV is assigned to provide communication supports for four UGVs sufficiently.

Additionally, we compared the MQTT and conventional HTTP based communication protocol in the context of the proposed framework. HTTP is document-centric, while MQTT is a data-centric protocol, meaning that HTTP utilizes the physical location of the IP address, whereas MQTT uses topics [12]. As a result, MQTT is expected to be lighter, faster and more efficient than HTTP [12]. In order to conduct a quantitative evaluation, the payload of the communication sequences is recorded for MQTT and HTTP-based

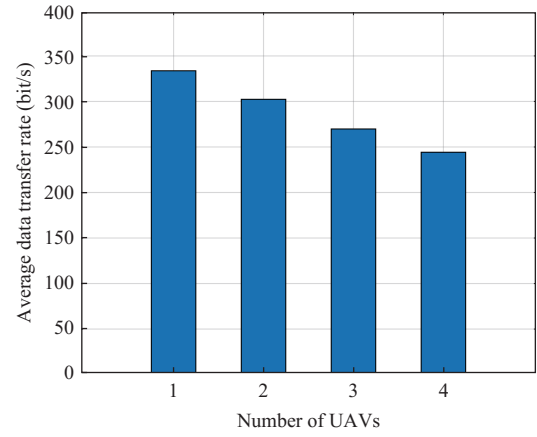


Fig. 11. Data transfer-rate for different number of UAVs.

communication. Based on the recorded data and Wireshark evaluation output, on average HTTP protocol-based data exchange takes 1164 bytes of traffic for publishing a single message, whereas MQTT protocol takes only 200 bytes. Unlike HTTP, MQTT protocol can use an established communication link between two nodes for multiple sessions, which further reduces the average time per message. As shown in Fig. 12, MQTT-based communication was able to send up to five messages using one established connection, while HTTP-based communication created a new connection to publish each new message. This confirms that the use of MQTT protocol is beneficial to make the proposed system efficient.

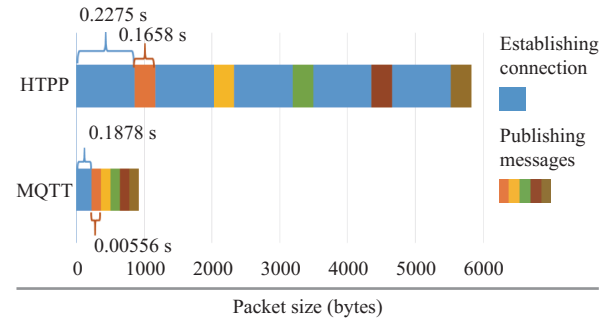


Fig. 12. MQTT and HTTP efficiency comparison.

Furthermore, the performance of the built antenna tracker has been evaluated using network analyzer instrument. The experiment is conducted by transmitting signal at 2.4 GHz frequency with various transmitting power and measuring the received signal using Agilent network analyzer. As shown in Fig. 13, the measured results are satisfactory and close to the theoretical values as calculated by Friss transmission equation $P_r = P_t G_t G_r \left(\frac{\lambda}{4\pi R} \right)^2$, where P_t denotes transmission power, G_t and G_r represent the transmitter and receiver antenna gains, respectively. λ is the wavelength of the radio frequency and R is the distance between the antenna tracker and UAV.

The antenna tracker's operating range for different input power is compared with the case when no antenna tracker is used, as shown in Fig. 14. It can be seen that the antenna tracker improves the framework range significantly. For

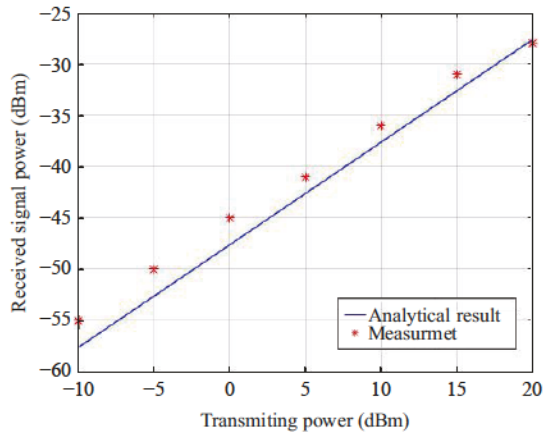


Fig. 13. Antenna tracker measurements compared with theoretical vlaues.

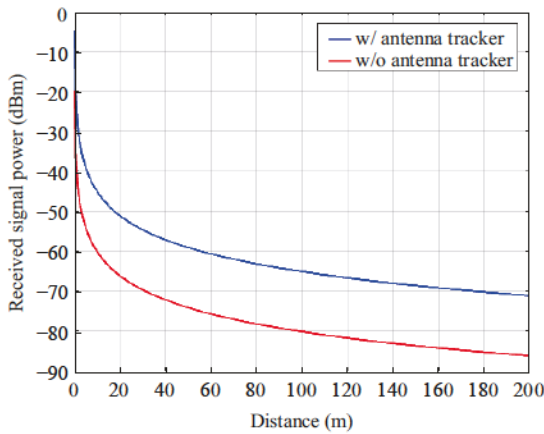


Fig. 14. Impact of antenna tracker in improving the range of the framework.

example, for the applications that require at least 70 dBm received signal power, the antenna tracker improves the range of the system from 60 m to 160 m, which is 166.6% increment.

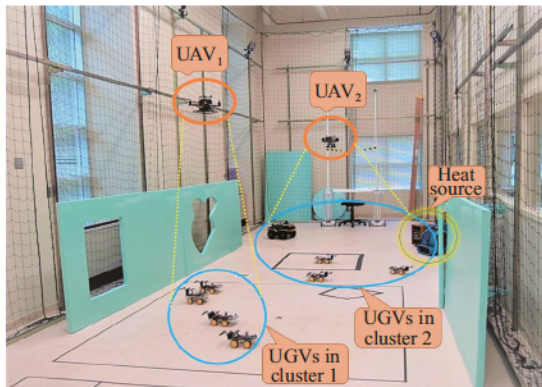
C. Control System Evaluation

Two scenarios are set up to verify and evaluate the collaborative control between the entities in our system. Prior

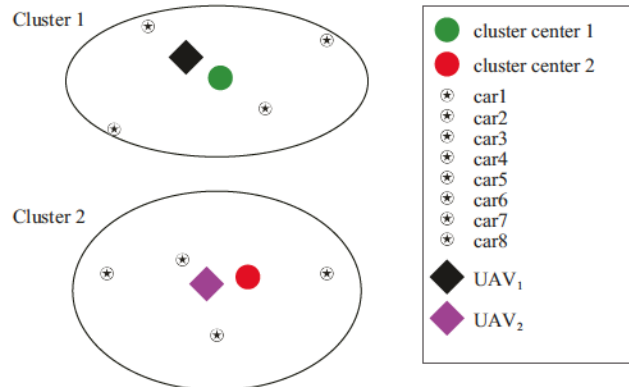
to implementing the scenarios, the developed UAVs have been tested in the lab environment for different low-level states such as hold, takeoff, hover, search and land [54]. The first scenario is shown in Fig. 15. In this scenario, a collaborative search and rescue operation that involves UGVs, UAVs, and RCS is verified to assist and accelerate the search process. This scenario assimilates a fire situation, where the UGVs and UAVs are assigned to search for the heat or the light source. In order to accomplish this mission, the UGVs and UAVs autonomously work together to locate a given target. As shown in Fig. 15(a), eight UGVs operating under two UAVs, the UGVs continuously monitor the environment and send the sensor reading data to the UAV and the UAVs forward it to the RCS. Fig. 15(b), shows that the UAVs are tracking the center of two groups of UGVs for hovering at the optimal position to provide maximum network coverage. In the RCS, the collected data is displayed in real-time. When there is a sudden change in sensor reading, it can easily be seen by the human operator, which is important to take an immediate measure.

As a sample application of the proposed framework, the second scenario was created to test the capability of the UAV's controller to autonomously track a moving UGV target and landing on top of it as shown in Fig. 16(a). The objectives of this scenario are to recharge UAVs and transport them to a distant location when required. We used two UAVs and two Jackal UGVs as moving targets. A flat roof is attached to the top of each Jackal UGV to create enough space for UAVs' landing. The UAVs take off from the ground and hovers on top of the two moving Jackals. Each UAV follows a Jackal by tracking its position, and then the UAV autonomously lands on top of the Jackal. To track the Jackals, the UAV selects the closest one by calculating the distance of the Jackals from its initial position. The following explains two major parts of this scenario, i.e., tracking and landing, in detail.

1) *Tracking*: To implement the tracking, we used the VICON motion capture system in the lab environment. The VICON system provides the current (x, y) coordinate of the UGVs' with respect to a reference global coordinate frame (which is calibrated previously). This position information is used by the UAVs' position controller as the target position.



(a) Demonstration of a collaborative search and rescue operation using UGVs and UAVs.



(b) The clustering algorithm running in real-time and sending signal to the UAVs to hover at the center of group of UGVs.

Fig. 15. UAV, UGV, and RCS based scenario demonstration.

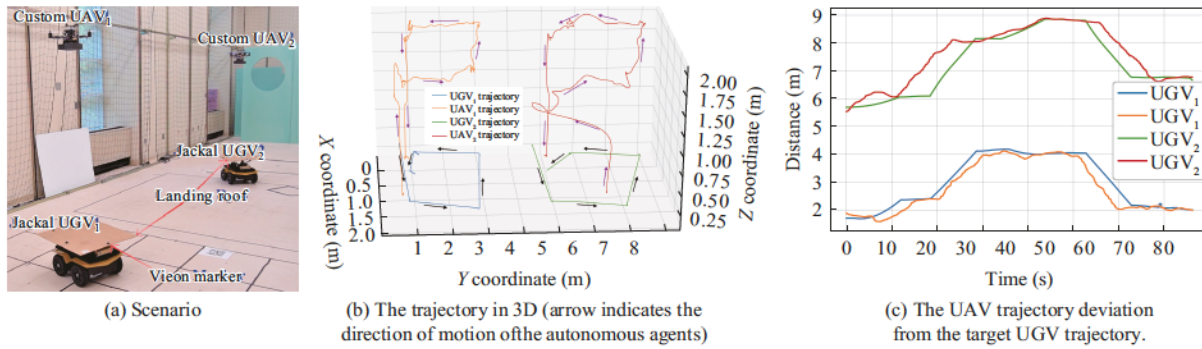


Fig. 16. Autonomous landing of UAVs on UGV targets and trajectories of the autonomous agents for the whole scenario.

We used a PID controller to control the position of the UAVs. The tuned horizontal and vertical position controller gains are K_{ph} (proportional gain) = 0.95, K_{Ih} (integral gain) = 0, K_{Dh} (derivative gain) = 0, K_{pv} = 1.0, K_{Iv} = 0.02, and K_{Dv} = 0. Therefore, when the positions of the UGVs are changed, the target positions of the UAVs' position controllers are also changed. While tracking, the altitudes of the UAVs' are kept constant. It is assumed that there is no obstacle in the flying zone. Therefore, the UAVs always follow a straight line trajectory from its current position to the target position.

2) *Landing*: The landing approach is adopted from [66]. In this approach, the UAV sends a signal to the UGV and observes the position of the UGV for a given time period (10 s for our implementation). If the position of the UGV does not change for 10 s, the UAV starts to descend with a speed of 0.7 ms^{-1} . If the UGV does not stop moving it sends another signal. To confirm the landing on the platform, we used two conditions: 1) the difference in Z coordinate of the UAV and UGV must be less than a threshold (in the experiment, it was 25 cm which is the height of the UAV); and 2) zero linear velocity reading from the IMU sensor of the UAV for a given period (10 s). The first condition ensures that the UAV and the UGV are in contact and the second condition implies that the UAV stops moving.

The recorded trajectory in Fig. 16(b), shows that as soon as the UAV takes off, it locates the target UGV based on its position and then lands on the top of the UGV. Moreover, the deviation of the UAV trajectories from the target UGV trajectories are measured from the logged sensor data and are shown in Fig. 16(c). The average calculated deviations are 15.57 cm for UAV₁ and 17.34 cm for UAV₂.

VII. CONCLUSION

In this paper, a collaborative wireless autonomous systems network (WASN) framework for disaster area management has been proposed. The framework enables synergy between heterogeneous autonomous system for assisting human agents to safely investigate post-disaster areas in a timely manner. The framework has been implemented using UAVs, UGVs, antenna tracker, cloud-based RCS, and IoT-based communication protocol. The system components have been developed using low-cost, off-the-shelf materials and open-source software. AADL formal method has been used to analyze the end-to-end latency of the overall system. The hardware implementation of the proposed framework has been

tested under different scenarios and the associated performances were evaluated from three perspectives, namely, AADL verification, communication and control systems. The measurements verified that the developed system meets the AADL timing analysis requirements. Experimental results show that the MQTT communication protocol is more efficient for our application compared to other conventional protocols. Furthermore, the evaluation results of the control system tested under different scenarios shows the effectiveness of the implemented control mechanisms. Future work will enhance the framework by implementing computer vision assisted control mechanism and machine learning enabled object segmentation technique for better monitoring of unreachable disaster areas from the aerial view.

ACKNOWLEDGMENT

The authors would like to thank the NASA Langley Research Center, Air Force Research Laboratory, Office of the Secretary of Defense (OSD), and the National Science Foundation (NSF) for support of this research.

REFERENCES

- [1] P. Pace, G. Aloï, G. Caliciuri, and G. Fortino, "A mission-oriented coordination framework for teams of mobile aerial and terrestrial smart objects," *Mobile Netw. Appl.*, vol. 21, no. 4, pp. 708–725, Aug. 2016.
- [2] O. Briante, V. Loscri, P. Pace, G. Ruggeri, and N. R. Zema, "Comvivor: An evolutionary communication framework based on survivors' devices reuse," *Wirel. Per. Commun.*, vol. 85, no. 4, pp. 2021–2040, Dec. 2015.
- [3] M. Erdelj and E. Natalizio, "UAV-assisted disaster management: Applications and open issues," in *Proc. Int. Conf. Computing, Networking and Communications*, Kauai, USA, 2016, pp. 1–5.
- [4] K. Nagatani, S. Kiribayashi, Y. Okada, K. Otake, K. Yoshida, S. Tadokoro, T. Nishimura, T. Yoshida, E. Koyanagi, M. Fukushima, and S. Shinji Kawatsuma, "Gamma-ray irradiation test of electric components of rescue mobile robot quince," in *Proc. IEEE Int. Symp. Safety, Security, and Rescue Robotics*, Kyoto, Japan, 2011, pp. 56–60.
- [5] M. Li, K. J. Lu, H. Zhu, M. Chen, S. W. Mao, and B. Prabhakaran, "Robot swarm communication networks: Architectures, protocols, and applications," in *Proc. 3rd Int. Conf. Communications and Networking in China*, Hangzhou, China, 2008, pp. 162–166.
- [6] I. Kassabalidis, M. A. El-Sharkawi, R. J. Marks, P. Arabshahi, and A. A. Gray, "Swarm intelligence for routing in communication networks," in *Proc. IEEE Global Telecommunications Conf.*, San Antonio, USA, 2001, pp. 3613–3617.
- [7] S. B. Hadj, S. Rekhis, N. Boudriga, and A. Bagula, "A cloud of UAVs for the delivery of a sink as a service to terrestrial WSNs," in *Proc. 14th Int. Conf. Advances in Mobile Computing and Multi Media*, Singapore, 2016, pp. 317–326.
- [8] V. Sharma and R. Kumar, "Cooperative frameworks and network

- models for flying ad hoc networks: A survey,” *Concurr. Comput. Pract. Exp.*, vol. 29, no. 4, pp. e3931, Feb. 2017.
- [9] X. Li, D. N. Guo, H. R. Yin, and G. Wei, “Drone-assisted public safety wireless broadband network,” in *Proc. IEEE Wireless Communications and Networking Conf. Workshops*, New Orleans, USA, 2015, pp. 323–328.
 - [10] N. Bezzo, B. Griffin, P. Cruz, J. Donahue, R. Fierro, and J. Wood, “A cooperative heterogeneous mobile wireless mechatronic system,” *IEEE/ASME Trans. Mech.*, vol. 19, no. 1, pp. 20–31, Feb. 2012.
 - [11] F. Flammini, C. Pragliola, and G. Smarra, “Railway infrastructure monitoring by drones,” in *Proc. Int. Conf. Electrical Systems for Aircraft, Railway, Ship Propulsion and Road Vehicles & Int. Transportation Electrification Conf.*, Toulouse, France, 2016, pp. 1–6.
 - [12] T. Yokotani and Y. Sasaki, “Comparison with HTTP and MQTT on required network resources for IoT,” in *Proc. Int. Conf. Control, Electronics, Renewable Energy and Communications*, Bandung, Indonesia, 2016, pp. 1–6.
 - [13] N. Naik, “Choice of effective messaging protocols for IoT systems: MQTT, CoAP, AMQP and HTTP,” in *Proc. IEEE Int. Systems Engineering Symp.*, Vienna, Austria, 2017, pp. 1–7.
 - [14] H. J. Yu and Y. Yang, “Latency analysis of automobile ABS based on AADL,” in *Proc. Int. Conf. Industrial Control and Electronics Engineering*, Xi’an, China, 2012, pp. 1835–1838.
 - [15] M. Munoz, “Space systems modeling using the architecture analysis & design language (AADL),” in *Proc. IEEE Int. Symp. Software Reliability Engineering Workshops*, Pasadena, USA, 2013, pp. 97–98.
 - [16] T. Zhang, Y. C. Jiang, J. D. Ye, C. Jing, and H. M. Qu, “An AADL model-based safety analysis method for flight control software,” in *Proc. Int. Conf. Computational Intelligence and Communication Networks*, Bhopal, India, 2014, pp. 1148–1152.
 - [17] J. Penders, L. Alboul, U. Witkowski, A. Naghsh, J. Saez-Pons, S. Herbrechtsmeier, and M. El-Habbal, “A robot swarm assisting a human fire-fighter,” *Adv. Robot.*, vol. 25, no. 1-2, pp. 93–117, 2011.
 - [18] R. Ventura and P. U. Lima, “Search and rescue robots: The civil protection teams of the future,” in *Proc. 3rd Int. Conf. Emerging Security Technologies*, Lisbon, Portugal, 2012, pp. 12–19.
 - [19] A. Ollero, J. Alcazar, F. Cuesta, F. Lopez-Pichaco, and C. Nogales, “Helicopter teleoperation for aerial monitoring in the comets multi-UAV system,” in *Proc. 3rd IARP Workshop on Service, Assistive and Personal Robots*, Madrid, Spain, 2003.
 - [20] J. H. Hong, B. C. Min, J. M. Taylor, V. Raskin, and E. T. Matson, “NL-based communication with firefighting robots,” in *Proc. IEEE Int. Conf. Systems, Man, and Cybernetics*, Seoul, South Korea, 2012, pp. 1461–1466.
 - [21] L. Gupta, R. Jain, and G. Vaszkun, “Survey of important issues in UAV communication networks,” *IEEE Communications Surveys & Tutorials*, vol. 18, no. 2, pp. 1123–1152, 2016.
 - [22] E. P. De Freitas, T. Heimfarth, I. F. Netto, C. E. Lino, C. E. Pereira, A. M. Ferreira, F. R. Wagner, and T. Larsson, “UAV relay network to support WSN connectivity,” in *Proc. Int. Congr. Ultra Modern Telecommunications and Control Systems*, Moscow, Russia, 2010, pp. 309–314.
 - [23] W. T. L. Teacy, J. Nie, S. McClean, and G. Parr, “Maintaining connectivity in UAV swarm sensing,” in *Proc. IEEE Globecom Workshops*, Miami, USA, 2010, pp. 1771–1776.
 - [24] D. Orfanus, E. P. de Freitas, and F. Eliassen, “Self-organization as a supporting paradigm for military UAV relay networks,” *IEEE Commun. Lett.*, vol. 20, no. 4, pp. 804–807, Apr. 2016.
 - [25] G. Fortino, W. Russo, C. Savaglio, W. M. Shen, and M. C. Zhou, “Agent-oriented cooperative smart objects: From IoT system design to implementation,” *IEEE Trans. Syst. Man Cybern. Syst.*, vol. 48, no. 11, pp. 1939–1956, Nov. 2018.
 - [26] C. Savaglio, M. Ganzha, M. Paprzycki, C. Bădică, M. Ivanović, and G. Fortino, “Agent-based internet of things: State-of-the-art and research challenges,” *Future Generat. Comput. Syst.*, vol. 102, pp. 1038–1053, Jan. 2020.
 - [27] K. Sundaresan, E. Chai, A. Chakraborty, and S. Rangarajan, “Skylite: End-to-end design of low-altitude UAV networks for providing LTE connectivity,” arXiv preprint arXiv: 1802.06042, 2018.
 - [28] S. A. R. Naqvi, S. A. Hassan, H. Pervaiz, and Q. Ni, “Drone-aided communication as a key enabler for 5G and resilient public safety networks,” *IEEE Commun. Mag.*, vol. 56, no. 1, pp. 36–42, Jan. 2018.
 - [29] W. S. Shi, H. B. Zhou, J. L. Li, W. C. Xu, N. Zhang, and X. M. Shen, “Drone assisted vehicular networks: Architecture, challenges and opportunities,” *IEEE Netw.*, vol. 32, no. 3, pp. 130–137, May/Jun. 2018.
 - [30] A. Girma, X. Y. Yan, and A. Homaifar, “Driver identification based on vehicle telematics data using LSTM-recurrent neural network,” arXiv preprint arXiv: 1911.08030, 2019.
 - [31] A. Girma, S. Amsalu, A. Workneh, M. Khan, and A. Homaifar, “Deep learning with attention mechanism for predicting driver intention at intersection,” arXiv preprint arXiv: 2006.05918, 2020.
 - [32] S. W. Zhang, Y. Zeng, and R. Zhang, “Cellular-enabled UAV communication: A connectivity-constrained trajectory optimization perspective,” *IEEE Trans. Commun.*, vol. 67, no. 3, pp. 2580–2604, Mar. 2019.
 - [33] A. Orsino, A. Ometov, G. Fodor, D. Moltchanov, L. Militano, S. Andreev, O. N. C. Yilmaz, T. Tirronen, J. Torsner, G. Araniti, A. Iera, M. Dohler, and Y. Koucheryavy, “Effects of heterogeneous mobility on D2D- and drone-assisted mission-critical MTC in 5G,” *IEEE Commun. Mag.*, vol. 55, no. 2, pp. 79–87, Feb. 2017.
 - [34] D. Wu, X. Sun, and N. Ansari, “An FSO-based drone assisted mobile access network for emergency communications,” *IEEE Trans. Netw. Sci. Eng.*, 2019. DOI: 10.1109/TNSE.2019.2942266.
 - [35] A. Trotta, M. Di Felice, F. Montori, K. R. Chowdhury, and L. Bononi, “Joint coverage, connectivity, and charging strategies for distributed UAV networks,” *IEEE Trans. Robot.*, vol. 34, no. 4, pp. 883–900, Aug. 2018.
 - [36] I. Kovacs, R. Amorim, H. C. Nguyen, J. Wigard, and P. Mogensen, “Interference analysis for UAV connectivity over LTE using aerial radio measurements,” in *Proc. IEEE 86th Vehicular Technology Conf.*, Toronto, Canada, 2017, pp. 1–6.
 - [37] O. S. Oubbati, A. Lakas, F. Zhou, M. Güneş, N. Lagraa, and M. B. Yagoubi, “Intelligent UAV-assisted routing protocol for urban VANETs,” *Comput. Commun.*, vol. 107, pp. 93–111, Jul. 2017.
 - [38] X. Y. Yan, M. Razeghi-Jahromi, A. Homaifar, B. A. Erol, A. Girma, and E. Tunstel, “A novel streaming data clustering algorithm based on fitness proportionate sharing,” *IEEE Access*, vol. 7, pp. 184985–185000, Jun. 2019.
 - [39] G. Fortino, C. Savaglio, C. E. Palau, J. S. de Puga, M. Ganzha, M. Paprzycki, M. Montesinos, A. Liotta, and M. Llop, “Towards multi-layer interoperability of heterogeneous IoT platforms: The INTER-IOT approach,” in *Integration, Interconnection, and Interoperability of IoT Systems*, R. Gravina, C. E. Palau, M. Manso, A. Liotta, and G. Fortino, Eds. Cham, Germany: Springer, 2018, pp. 199–232.
 - [40] N. H. Motlagh, T. Taleb, and O. Arouk, “Low-altitude unmanned aerial vehicles-based internet of things services: Comprehensive survey and future perspectives,” *IEEE Internet Things J.*, vol. 3, no. 6, pp. 899–922, Dec. 2016.
 - [41] Y. Zhou, N. Cheng, N. Lu, and X. S. Shen, “Multi-UAV-aided networks: Aerial-ground cooperative vehicular networking architecture,” *IEEE Veh. Technol. Mag.*, vol. 10, no. 4, pp. 36–44, Dec. 2015.
 - [42] J. P. Dash, G. D. Pearce, and M. S. Watt, “UAV multispectral imagery can complement satellite data for monitoring forest health,” *Remote Sens.*, vol. 10, no. 8, pp. 1216, Aug. 2018.
 - [43] N. Goddemeier, K. Daniel, and C. Wietfeld, “Role-based connectivity management with realistic air-to-ground channels for cooperative UAVs,” *IEEE J. Sel. Areas Commun.*, vol. 30, no. 5, pp. 951–963, Jun. 2012.
 - [44] Y. Ham, K. K. Han, J. J. Lin, and M. Golparvar-Fard, “Visual monitoring of civil infrastructure systems via camera-equipped unmanned aerial vehicles (UAVs): A review of related works,” *Vis. Eng.*, vol. 4, no. 1, pp. 1–8, 2016.
 - [45] C. B. Luo, J. Nightingale, E. Asemota, and C. Grecos, “A UAV-cloud system for disaster sensing applications,” in *Proc. IEEE 81st Vehicular Technology Conf.*, Glasgow, UK, 2015, pp. 1–5.
 - [46] Jackal unmanned ground vehicle. [Online]. Available: <https://clearpathrobotics.com/jackal-small-unmanned-ground-vehicle>.

- [47] Z. Beni c, P. Piljek, and D. Kotarski, "Mathematical modelling of unmanned aerial vehicles with four rotors," *Interdiscip. Descript. Complex Syst.*, vol. 14, no. 1, pp. 88–100, Jan. 2016.
- [48] M. Behniapoor, Z. Yuan, A. Hailemichael, K. Vinh, B. Bowles, A. Karimodini, and A. Homaifar, "Development of a micro aerial vehicle," in *Proc. World Automation Congr.*, Rio Grande, Puerto Rico, 2016, pp. 1–6.
- [49] J. Baichtal, *Building Your Own Drones: A Beginners' Guide to Drones, UAVs, and ROVs*. Indianapolis, Indiana: Que Publishing, 2015.
- [50] G. H. T. Navajas and S. R. Prada, "Building your own quadrotor: A mechatronics system design case study," in *Proc. III Int. Congr. Engineering Mechatronics and Automation*, Cartagena, Colombia, 2014, pp. 1–5.
- [51] A. Mashood, M. Mohammed, M. Abdulwahab, S. Abdulwahab, and H. Noura, "A hardware setup for formation flight of UAVs using motion tracking system," in *Proc. 10th Int. Symp. Mechatronics and its Applications*, Sharjah, United Arab Emirates, 2015, pp. 1–6.
- [52] S. Al Habsi, M. Shehada, M. Abdoon, A. Mashood, and H. Noura, "Integration of a vicon camera system for indoor flight of a parrot AR drone," in *Proc. 10th Int. Symp. Mechatronics and its Applications*, Sharjah, United Arab Emirates, 2015, pp. 1–6.
- [53] N. Michael, D. Mellinger, Q. Lindsey, and V. Kumar, "The GRASP multiple micro-UAV testbed," *IEEE Robot. Automat. Mag.*, vol. 17, no. 3, pp. 56–65, Sept. 2010.
- [54] M. Sarkar, A. Homaifar, B. A. Erol, M. Behniapoor, and E. Tunstel, "PIE: A tool for data-driven autonomous UAV flight testing," *J. Intell. Robot. Syst.*, vol. 98, no. 2, pp. 421–438, May 2020.
- [55] M. R. Kusanovic and M. K. Stojcev, "Connecting wireless sensor networks to internet," *Facta Univ.-Ser. Mech. Eng.*, vol. 9, no. 2, pp. 169–182, 2011.
- [56] U. Hunkeler, H. L. Truong, and A. Stanford-Clark, "MQTT-s—a publish/subscribe protocol for wireless sensor networks," in *Proc. 3rd Int. Conf. Communication Systems Software and Middleware and Workshops*, Bangalore, India, 2008, pp. 791–798.
- [57] L. Durkop, B. Czybik, and J. Jasperneite, "Performance evaluation of M2M protocols over cellular networks in a lab environment," in *Proc. 18th Int. Conf. Intelligence in Next Generation Networks*, Paris, France, 2015, pp. 70–75.
- [58] L. Mainetti, L. Patrono, and A. Vilei, "Evolution of wireless sensor networks towards the internet of things: A survey," in *Proc. 19th Int. Conf. Software, Telecommunications and Computer Networks*, Split, Croatia, 2011, pp. 1–6.
- [59] A. A. Dahoud and M. Fezari, "NodeMCU V3 For fast IoT application development," [Online]. Available: https://paesaggiaperti.org/alfonso/rioba_tech/raw/branch/master/piattaforma/NodeMCUV3.pdf.
- [60] P. H. Feiler, B. A. Lewis, and S. Vestal, "The SAE architecture analysis & design language (AADL) a standard for engineering performance critical systems," in *Proc. IEEE Conf. Computer Aided Control System Design, IEEE Int. Conf. Control Applications, IEEE Int. Symp. Intelligent Control*, Munich, Germany, 2006, pp. 1206–1211.
- [61] Espressif Systems. Esp8266ex datasheet. [Online]. Available: https://cdn-shop.adafruit.com/product-files/2471/0A-ESP8266_Datasheet_EN_v4.3.pdf.
- [62] R. P. Foundation. Raspberry pi 3 model B+. [Online]. Available: <https://static.raspberrypi.org/files/product-briefs/Raspberry-Pi-Model-Bplus-Product-Brief.pdf>.
- [63] Moog Components Group. DBH-0472 models. [Online]. Available: https://www.moog.com/content/dam/moog/literature/MCG/DBH-0472_Model_DTS.pdf.
- [64] MPJA. HC-SR04 user guide. [Online]. Available: https://www.mpja.com/download/hc-sr04_ultrasonic_module_user_guidejohn.pdf.
- [65] A. Sikora and V. F. Groza, "Fields tests for ranging and localization with time-of-flight-measurements using chirp spread spectrum RF-devices," in *Proc. IEEE Instrumentation & Measurement Technology Conf.*, Warsaw, Poland, 2007, pp. 1–6.
- [66] P. R. Palafox, M. Garzon, J. Valente, J. J. Rold n, and A. Barrientos, "Robust visual-aided autonomous takeoff, tracking, and landing of a small UAV on a moving landing platform for life-long operation," *Appl. Sci.*, vol. 9, no. 13, pp. 2661, 2019.



Abenezer Girma (S'18) received the B.Sc. degree in electrical and electronics engineering from Addis Ababa University, The Federal Democratic Republic of Ethiopia, in 2016. He worked as a digital communication Officer at Total S.C Ethiopia. He is currently pursuing the Ph.D. degree in electrical and computer engineering at North Carolina A&T State University, USA. He is also a graduate research Assistant at Autonomous Control Information and Technology (ACIT) Institute in NCAT. He is the

Recipient of National Aerospace Institute Fellowship, where he is working on addressing computer vision challenges for future unmanned aerial system (UAS). His research interests include deep learning, efficient convolution neural network architecture design, computer vision, autonomous system, internet of things (IoT), and data analysis.



Niloofar Bahadori received the B.Sc. degree in electrical and electronics engineering from the Isfahan University, Iran, in 2011, the M.Sc. degree (with Hons.) in electrical and radio frequency (RF) engineering from the Semnan University, Iran, in 2013. She is currently pursuing the Ph.D. degree at North Carolina A&T State University, USA. Her current research interests include device-to-device (D2D) and machine-to-machine (M2M) communication, mm wave band communication, internet of things (IoT), the applications of machine learning in wireless network, and game theory. She is the Recipient of the 2019 IEEE Wireless Telecommunications Symposium (WTS) Best Paper Award.



Mrinmoy Sarkar received the B.S. degree in electrical and electronic engineering from Bangladesh University of Engineering and Technology in 2016. He is currently pursuing the Ph.D. degree in electrical and computer engineering at North Carolina A&T State University, USA. He worked as a Software Engineer at Samsung R&D Institute Bangladesh Ltd. from July 2016–July 2017. His research interests include testing and evaluation of autonomous behavior of UAV agents using machine learning technique, developing different complex scenarios for testing UAVs and a heterogeneous system consisting of UAVs as well as UGVs, analyzing the behavior of large-scale autonomous systems and the application of machine learning techniques in robotics. He is a Member of IEEE and Golden Key International Honour Society.



Tadewos G. Tadewos received the bachelor degree in computer and electrical engineering from Addis Ababa University, The Federal Democratic Republic of Ethiopia, in 2010. He received the M.Sc degree in embedded systems from Polytecnico di Torino, Italy, in 2014. In 2016, he joined the North Carolina A&T State University to pursue the Ph.D. degree. His research interests include testing and evaluation of autonomous vehicles, formal verification, tasking of multi-agent systems, and aerial and ground robotics. He is a Member of Autonomous Cooperative Control of Emergent Systems of Systems (ACCESS) Lab and Testing, Evaluation and Control of Heterogeneous Large-Scale Systems of Autonomous Vehicles (TECHLAV) Center.



Mohammad R. Behnia received the B.Sc. degree in electrical and electronics engineering from the Tehran Polytechnic University, Iran, and the M.Sc. degree in autonomous and control from the North Carolina A&T State University, USA, in 2017. While pursuing the Ph.D. studies, he joined Autonomous Control and Information Technology Institute as a Lab Manager. He is currently a Research Operation Manager with the Department of Electrical and Computer Engineering at North Carolina A&T State University. His research interests include autonomous control, aerial and ground vehicle, considering UAV applications.



Mahmoud Nabil is an Assistant Professor in the Department of Electrical and Computer Engineering, North Carolina A&T State University, USA. He received the Ph.D. degree in electrical and computer engineering from Tennessee Tech University, USA, in August 2019. He received the B.S. degree and the M.S. degree with honors in computer engineering from Cairo University, Egypt, in 2012 and 2016, respectively. He published many journals and conferences in different prestigious venues such as *IEEE Internet of Things Journal*, *IEEE Transactions of Dependable and Secure Computing*, *IEEE Access*, *International Conference on Communication (ICC)*, *International Conference on Pattern Recognition (ICPR)*, and *International Conference on Wireless Communication (WCNC)*. His research interests include security and privacy in smart grid, machine learning applications, vehicular Ad-hoc networks, and blockchain applications.



Ali Karimoddini (SM'15) is an Associate Professor in the ECE Department at North Carolina A&T State University, USA. He received the bachelor degree in electrical and electronics engineering from the Amirkabir University of Technology, Iran, in 2003. He received the master degree in science in instrumentation and automation engineering from Petroleum University of Technology, Iran, in 2007. In 2008, he joined the National University of Singapore, Singapore, to pursue the Ph.D. degree, and then he joined the University of Notre Dame, USA, to conduct his postdoctoral studies. His research interests include cyber-physical systems,

control and robotics, resilient control systems, flight control systems, multi-agent systems, and human-machine interactions. He is the Director of NC-CAV Center of Excellence on Advanced Transportation, Director of ACCESS Laboratory, and the Deputy Director of TECHLAV DoD Center of Excellence. His research has been supported by different federal funding agencies and industrial partners. He is a Senior Member of IEEE, and a Member of AIAA and ISA.



Abdollah Homaifar received the B.S. and M.S. degrees from the State University of New York at Stony Brook, USA, in 1979 and 1980, respectively, and the Ph.D. degree from the University of Alabama, USA, in 1987, all in electrical engineering. He is the NASA Langley Distinguished Professor and the Duke Energy Eminent Professor in the Department of Electrical and Computer Engineering at North Carolina A&T State University (NCA&TSU), USA. He is the Director of the Autonomous Control and Information Technology Institute and the Testing, Evaluation, and Control of Heterogeneous Large-Scale Systems of Autonomous Vehicles (TECHLAV) Center at NCA&TSU. His research interests include machine learning, unmanned aerial vehicles (UAVs), testing and evaluation of autonomous vehicles, optimization, and signal processing. He also serves as an Associate Editor of the *Journal of Intelligent Automation and Soft Computing*, and a Reviewer for *IEEE Transactions on Fuzzy Systems*, *Man Machines and Cybernetics*, and *Neural Networks*. He is a Member of the IEEE Control Society, Sigma Xi, Tau Beta Pi, and Eta Kappa Nu.