IEEE SA
STANDARDS ASSOCIATION

IEEE SA WHITE PAPER

IEEE P1920.2 WORKING GROUP

USE CASES FOR VEHICLE-TO-VEHICLE COMMUNICATIONS FOR UNMANNED AIRCRAFT SYSTEMS

Authored by

Kenneth Bandelier
VIAVI Solutions Application Engineer

Saba Al-Rubaye
Reader in Autonomous and Connected Systems, Cranfield University, United Kingdom

Marco Hernandez
Center for Wireless Communications, Oulu University, Finland
Yokosuka Research Park International Alliance Institute, Japan
National Institute of Information and Communications Technology, Japan

Kamesh Namuduri
Professor of Electrical Engineering, University of North Texas

Stefano Savazzi
Researcher, Consiglio Nazionale delle Ricerche (CNR), Institute of Electronics, Computer and Telecommunication Engineering (IEIIT)

IEEE

TRADEMARKS AND DISCLAIMERS

IEEE believes the information in this publication is accurate as of its publication date; such information is subject to change without notice. IEEE is not responsible for any inadvertent errors.

The ideas and proposals in this specification are the respective author's views and do not represent the views of the affiliated organization.

ACKNOWLEDGMENTS

Special thanks are given to the following contributors to this paper:

Gerhard Schauble – CEO - North American Aerospace
Ivan Petrunin – Centre for Autonomous and Cyber-Physical Systems School of Aerospace, Transport and Manufacturing Cranfield University, United Kingdom
Ashwin Ashok – Associate Professor, Department of Computer Science & Affiliate in Neuroscience, Georgia State University
R. Venkatesha Prasad – Associate Professor, EEMCS, TU Delft, The Netherlands
Shrikant Tangade – Associate Professor, CHRIST University, Kengeri Campus, Bangalore, India
Gary S. Griffith – Senior Enterprise Systems Architect, Information Technology & Data Analytics, The Boeing Company
Gürkan Gür – Senior Lecturer, Zurich University of Applied Sciences (ZHAW), Institute of Applied Information Technology (InIT), Switzerland
Sven Bilén – Professor, The Pennsylvania State University

The Institute of Electrical and Electronics Engineers, Inc. 3 Park Avenue, New York, NY 10016-5997, USA

PDF: STDVA26050 978-1-5044-9566-0

NOTICE AND DISCLAIMER OF LIABILITY CONCERNING THE USE OF IEEE SA DOCUMENTS

This IEEE Standards Association ("IEEE SA") publication ("Work") is not a consensus standard document. Specifically, this document is NOT AN IEEE STANDARD. Information contained in this Work has been created by, or obtained from, sources believed to be reliable, and reviewed by members of the activity that produced this Work. IEEE and the IEEE P1920.2 Working Group expressly disclaim all warranties (express, implied, and statutory) related to this Work, including, but not limited to, the warranties of: merchantability; fitness for a particular purpose; non-infringement; quality, accuracy, effectiveness, currency, or completeness of the Work or content within the Work. In addition, IEEE and the IEEE P1920.2 Working Group disclaim any and all conditions relating to: results; and workmanlike effort. This document is supplied "AS IS" and "WITH ALL FAULTS."

Although the IEEE P1920.2 Working Group members who have created this Work believe that the information and guidance given in this Work serve as an enhancement to users, all persons must rely upon their own skill and judgment when making use of it. IN NO EVENT SHALL IEEE-SA OR ICAP MEMBERS BE LIABLE FOR ANY ERRORS OR OMISSIONS OR DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO: PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS WORK, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE AND REGARDLESS OF WHETHER SUCH DAMAGE WAS FORESEEABLE.

Further, information contained in this Work may be protected by intellectual property rights held by third parties or organizations, and the use of this information may require the user to negotiate with any such rights holders in order to legally acquire the rights to do so, and such rights holders may refuse to grant such rights. Attention is also called to the possibility that implementation of any or all of this Work may require use of subject matter covered by patent rights. By publication of this Work, no position is taken by the IEEE with respect to the existence or validity of any patent rights in connection therewith. The IEEE is not responsible for identifying patent rights for which a license may be required, or for conducting inquiries into the legal validity or scope of patents claims. Users are expressly advised that determination of the validity of any patent rights, and the risk of infringement of such rights, is entirely their own responsibility. No commitment to grant licenses under patent rights on a reasonable or non-discriminatory basis has been sought or received from any rights holder.

This Work is published with the understanding that IEEE and the IEEE P1920.2 Working Group members are supplying information through this Work, not attempting to render engineering or other professional services. If such services are required, the assistance of an appropriate professional should be sought. IEEE is not responsible for the statements and opinions advanced in this Work.

TABLE OF CONTENTS

VEHICLE-TO-VEHICLE (V2V) UNMANNED AERIAL VEHICLE (UAV) COMMUNICATION BASED ON USE CASES

ABSTRACT

The proliferation of unmanned aerial vehicle (UAV)[1] usage in today's airspace creates unique challenges for coordinating movement of a multitude of different UAVs with different roles and responsibilities to ensure safe, secure, and efficient management in an ever-increasingly crowded airspace. The purpose of this paper is to look at the various use cases for UAVs and how these different cases impact the ability of UAVs to communicate with other UAVs. This paper expects UAVs to be complementary to ground communication (General Aviation Manufacturers Association [3], executive summary) to fulfill their operational missions. More specifically, this white paper examines how the detect-and-avoid or collision avoidance use case can be used to form the basis of a vehicle-to-vehicle (V2V) communication protocol and message format. The proposed solution is intended to meet the needs of regulatory requirements, such as the Federal Aviation Administration's Remote ID [2]), and other industry standards organizations, such as ASTM and RTCA, working on the Airborne Collision Avoidance System for Unmanned Aircraft (ACAS Xu) solution for UAVs, aerospace original equipment manufacturers (OEMs) building UAVs, and operators of UAVs. While security requirements and frequency usage were considered, the paper does not propose any detailed solutions in these areas as they will be offered by other sub-groups in the IEEE P1920.2 working group.

[1] UAV is considered "unmanned" in the context of this paper due to flying without a human pilot onboard, resulting in autonomous, semi-autonomous, or remotely piloted flight.

1. INTRODUCTION

The usage of UAVs has become more prevalent as the technology incorporated into them has become more proficient. The capabilities were initially pushed by military needs and have crossed over to the commercial realm. UAVs of today have greater capabilities and expanded missions as their technology and performance have increased. This has led to a natural desire for communication between UAVs so they can work safely and in a coordinated fashion to accomplish their respective missions.

The working group's first step in evaluating the possible communications between UAVs was looking at all possible uses for UAVs. These initial use cases where evaluated from the perspective of how they might impact the UAV's ability to communicate while being deployed. This allowed the group to evaluate what specific issues a UAV may encounter when attempting to communicate to other UAVs with respect to their use case scenarios.

It was quickly determined that the most prevalent communication usage would be collision avoidance or the detect-and-avoid scenario. This was deemed to be the primary driver of communications between vehicles. With this in mind, the operating environments were considered along with the unique challenges that each different environment may include. For instance, a UAV operating in the confines of a major city would be prone to encounter more stationary man-made obstructions and other moving objects at a greater density than would be encountered in a rural location, such as a farm. All scenarios have a need to detect and avoid obstacles in the UAV flight path and could benefit if one UAV could communicate this information to other UAVs to aid them in avoiding obstacles that they may not yet have detected.

The next step was to consider any proposed or existing regulations, such as the FAA's Remote ID regulation, and the minimum requirements needed to comply. The Remote ID regulation stipulates certain broadcast features, operating frequencies, and other specific informational data to be broadcast by UAVs. It also includes a distinction between UAVs operating under Code of Federal Regulations Title 14, Chapter I, Subchapter F, part 91 and part 107 [2]. The Remote ID regulation has a clear prohibition against usage of either ADS-B OUT or transponders for UAVs flying under part 107. Therefore, the working group proposed that if the V2V standard was compatible with Remote ID, it might be able to be used for a less-sophisticated form of collision avoidance. In any case, this means there are two distinct classes of UAVs: those operating with traditional transponder capability and those without it. The proposed communication solution is intended to be compatible with other proposals being produced by other standards issuing organizations.

The next step was to consider the role of various types of UAVs in operation and how the size of the craft would impact their capabilities. It was quickly surmised that the smallest UAVs would be limited in both capabilities and missions, being unable to match the capabilities of much larger UAVs. In addition, it was perceived that the capabilities and missions would scale up in proportion to the size of the craft. Larger UAVs would be able to carry more sensors, payload, and have better handling and range than smaller craft. The realization that craft size would dictate capabilities led to the development of categories or levels of UAVs.

The working group initially created three UAV levels based on vehicle size: small UAVs (less than 55 lb., no cargo, no passengers), medium UAVs (more than 55 lb. and less than general aviation aircraft), and large UAVs (equal to general aviation aircraft or larger). After further discussion, additional categories were added, such as vehicle type (fixed wing, rotorcraft, etc.) and vehicle usage (reconnaissance, cargo delivery, etc.), to provide a more fulsome description of the vehicle and its capabilities and limitations. It was determined that the safety and security requirements would be based on the UAV's mission or primary usage. For instance, more safety considerations would be required if cargo or passengers are being carried. The UAV communications are much more critical if the craft is a cargo UAV carrying valuable cargo, an air taxi loaded with passengers, or is acting as an emergency vehicle such as an air ambulance.

By setting up a UAV type designation based primarily on vehicle size, flight capabilities, and primary usage, it was determined that this would be useful in helping organize V2V communication but could also have the added benefit in situations where UAV congestion is heavy by allowing for prioritization of craft movements using the UAV type designation in conjunction with current position and flight path information.

This paper also looks at the different sensor and computational architectures that may be included on a UAV. The purpose is to highlight how the differences in capabilities, or the limitations due to a lack of certain sensors or computational architectures, may impact a UAVs role and/or ability to communicate to other vehicles. This effort is intended to provide guidance on minimum capabilities that the V2V communication standard should support.

Further topics discussed in the paper include the following:

- UAV sensors and how they impact UAV flight capabilities and performance, enable tasks, or affect mission complexity. Their output is integrated into V2V communications.
- UAV computational limitations due to craft size, system architecture and technology, and novel uses.
- UAV communication methods, message formats, and usage.

2. UAV SENSORS REQUIREMENTS

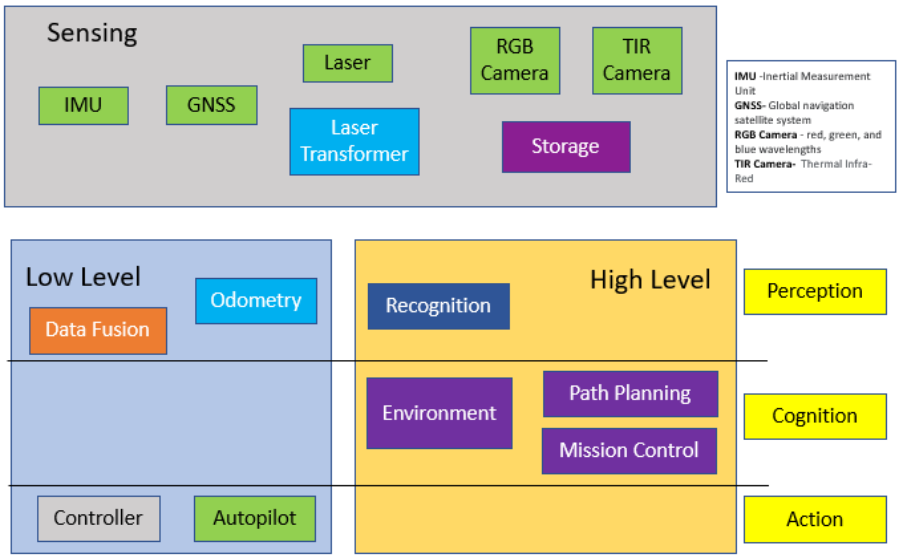
2.1. OVERVIEW

One of the key features of UAVs used in industry is the availability of a wide range of sensors with different functionalities for unmanned operations. Employing advanced sensors together with high-tech navigational and communication devices onboard a UAV makes it capable of carrying out different missions without risk to human operators. Apart from military use cases, UAVs are utilized for many commercial use cases such as parcel delivery, precision agriculture, fire detection and fighting, industrial inspection, maritime surveillance, and medical equipment transport. Considering the exponential increase in UAV applications in recent years, it is expected that UAVs will be a common element in airspace in near future.

2.2. UAV PAYLOAD SYSTEM ARCHITECTURE

UAVs are usually equipped with a set of monitoring and data-gathering sensors and devices such as cameras, infrared sensors, and thermal sensors that collect different information to be processed on board the UAV or transmitted to the ground base station for further analysis. The systems architecting process consists of aggregating, partitioning, integrating, and finally validating systems architecture. The architecting process is the one by which standards, protocols, rules, system structure, and interfaces are created in order to achieve the requirements of a system. In other words, it is the planning and building of structures to respond to a given need. For this purpose, first, the system components should be classified according to their degree of autonomy and cognitive functionality. From another perspective, the system components are categorized into low-level and high-level components, as shown in FIGURE 1. The low-level components are in charge of the perception and cognition, data fusion, and flight control of the UAV. These components can provide reliable autonomous flight and navigation, providing an interface between human operators and high-level components. In addition, situational awareness and mission-planning functionalities are provided via the high-level components.

FIGURE 1 **System architecture design**



In TABLE 1, several typical UAV applications and their corresponding data-communication requirements are listed. Although these applications were defined initially for UAVs, they can be carried out by urban air mobility (UAM) services as well. The requirements for passenger internet and infotainment communications (e.g., web browsing, video streaming, voice and video calls) would be similar to those for users of terrestrial networks.

TABLE 1 **Payload data communication requirement for typical UAV applications**

UAV application	Height coverage	Maximum payload traffic latency	Payload data rate (download/upload)
UAV delivery	100 m	500 ms	300 kbps/200 kbps
UAV filming	100 m	500 ms	300 kbps/30 Mbps
UAV fleet show	200 m	100 ms	200 kbps/200 kbps
Precision agriculture	300 m	500 ms	300 kbps/200 kbps
Search and rescue	100 m	500 ms	300 kbps/6 Mbps
Surveillance	100 m	3000 ms	300 kbps/10 Mbps
Infrastructure inspection	100 m	3000 ms	300 kbps/10 Mbps

2.3. UAV SENSORS

Measurements provided by onboard sensors are essential for UAVs and have a strong impact on their performance. All the measurement tools are mounted on the UAV to acquire detailed information at low altitudes. The UAV's navigation monitoring capabilities, with respect to several physical quantities in the environment around them, strictly depend on sensors measurement systems, as well as on data-processing techniques. The development of autonomous systems is very much connected to the ability of data analysis coming from the measurements provided by onboard sensors such as daylight and night vision cameras, LiDAR, radar, etc., in order to take advantage of their different positioning capability and to collect information related to different dimensions of the environments, as displayed in TABLE 2.

TABLE 2 UAV measurement with sensors

Technology	Estimated position and the actual one	Advantages
Camera	1 cm (in static test positions)	Good precision/Doppler effect, acoustic noise
Inertial measurement unit (**IMU**), optical, Global Navigation Satellite System (**GNSS**), ultrawideband (**UWB**)	10 cm	Good accuracy, but high cost

The onboard sensors have different applications, and depending on the UAV mission, the sensor payload of the UAV will vary. Due to the demand for UAVs in various applications—such as precision agriculture, search and rescue, wireless communications, and surveillance—several types of UAVs have been invented with different specifications for their size, weight, and range (see TABLE 3). They can be equipped with multiple sensors—including cameras, inertial measurement units (IMUs), light detection and ranging (LiDAR), and Global Positioning System (GPS)—to collect and transmit data in real-time. Examples of these sensors are as follows:

- LiDAR: range measurement, detection, 3D visualization
- High definition (HD) camera: visual inspection
- Multispectral camera: different wavelength imaging
- Short wave infrared (IR) camera: detection
- Ultrasonic: thickness measurement
- RADAR: detection
- Thermal camera: detection, inspection

TABLE 3 UAVs features for different UAV size use case

Features	Types of UAVs for intelligence, surveillance, and reconnaissance (ISR)		
	Small size, hobbyist	Mid-size, military and commercial	Large, military-specific
Payload	Limited	Moderate	Large
Endurance	Limited	Moderate	Long
Imaging	High-definition (HD) video and image	HD video and image and advanced radar	Advanced radar and electro-optical (EO)/IR imaging
Highlight features	GPS, waypoint navigation system	Target detection, encrypted datalink	Beyond line of sight (BLOS) operation, missiles release, autonomous mode

3. UAV COMPUTATIONAL REQUIREMENTS

3.1. OVERVIEW

UAVs are typically available in different sizes and specifications while the payload, namely the maximum weight that a UAV can carry (lifting capability), is typically adopted to classify computational requirements as well as processing power. Payloads of UAVs vary from a few pounds to hundreds of pounds. The larger the payload, the more processing power, equipment, and accessories can be carried at the expense of a larger UAV size, higher battery capacity, and shorter duration in the air. The following defines two main UAV classes, specifically:

- **< 55 lb. payload**: Small UAV devices typically perform simple operating tasks, such as autonomous conflict detection (and avoidance/resolution) and sensing. Small UAVs have slow speed (15 m/s maximum) and a flight time (endurance) of 20 min to 30 min.
- **> 55 lb. payload**: Large UAV devices can perform complex computations, e.g., serve as mobile edge servers, base stations, and access/coordination points, with a fusion of multiple sensors and complex autonomous maneuvering. Large UAVs might reach up to 100 m/s with an endurance of up to 4.5 hours.

UAVs rely on low-power single-board computers (SBCs) for their computational needs. In some cases, SBCs might be equipped with multiple tensor processing unit(s) or other specialized computing accelerators to improve processing power. This is a typical design for > 55 lb. UAVs. Simple UAVs (< 55 lb.) with reduced functions are often equipped with a system-on-a-chip (SoC) design.

3.2. COMPUTING HARDWARE AND EXAMPLES

TABLE 4 summarizes some examples of typical computing hardware and SBC suitable for integration on UAVs, as well as for onboard processing. For each model, we report the measured/expected GFLOP/sec, corresponding to 1 billion floating point operations per second. Processors suitable for UAVs might use varying computing architectures, typically ranging from Intel to ARM processor families. ARM processors are particularly attractive for UAV integration as they use less energy, thanks to their single-cycle computing set. However, their performance is lower than Intel processors and have a reduced operating temperature. Most notably, ARM (Cortex-A and Cortex-M) and ultra-low (U) power Intel processor families are typical choices as optimized for cost and energy-efficient controllers. In particular, the M series ARM CPUs have relatively smaller instruction set than A series, and often no floating-point unit. They are optimized for low cost and simple UAV tasks rather than high performance: therefore, they are a reasonable choice for UAVs less than 55 lb. UAVs, typically controlled from a ground station, combined with flash, random access memory (RAM), and peripherals. The A series ARM CPUs have a larger instruction set, a floating-point unit, memory management unit, and cache(s). They are optimized for high-performance UAV tasks and are a good choice for UAVs more than 55 lb., featuring considerably higher processing power. They also run an operating system (OS), often Linux based, as well as application programs.

Autonomously flying UAVs mainly rely on vision algorithms that combine multiple sensing data, ranging from video, light detection, and radio frequency (RF). When onboard processing is unfeasible, computing is offloaded to a ground station (see 3.3) that is used to process the sensor data and control/steer the UAV. In case onboard processing is employed, algorithms should be designed as sufficiently lightweight. Both large and small payload UAVs are thus often equipped with an additional tensor processing unit (TPU), namely a graphics processing unit (GPU), for efficient data processing and transfer of spatially coordinated image/sensor data throughout the system. Main requirements of onboard GPU are as follows:

- Energy efficiency, or the UAV flight time, typically the time required to deplete the 75% of the battery capacity
- Computing frequency, typically 650 MHz or above
- Throughput, often expressed in GFLOPS, frame per second (fps), or Gpixel/s, typically > 10 Gpixel/s

Finally, considering memory, most (i.e., open) firmware configurations currently available exceed 1 MB in size. Therefore, some simple SBC components and autopilots may not have enough flash memory to store the full firmware: a minimum size of 2 Mb is thus considered practical. Similar limitations also apply to RAM size.

Due to the limited computing ability of a single UAV, multiple interconnected UAVs could be considered to

simultaneously provide complex computing service that resort to cooperation or more elaborate task offloading. Such computing architectures supporting offloading and UAV cooperation are detailed in the next section.

TABLE 4 Examples of SOC/TPU processing suitable for UAVs

CPU only (SOC)	Computing accelerator: GPU/TPU	Examples
AM335x ARM Cortex A8, 1 GHz (0.055 GFLOPS)	No	BeagleBone/Beagleboard
ARM Cortex M4 core	No	Cube autopilot, Pixhawk
STM32H743 (2 Mb flash), ARM Cortex M7 core	No	Holybro (Durandal), CubePilot
STM32F765: ARM Cortex M7 core with DPFPU	No	
RK3288: ARM Cortex A17, 4 cores, up to 1.8 GHz	ARM Mali-T764 GPU, quad core 3D graphics	DJI and other models
216 MHz/512 kb RAM/2 MB flash	Not applicable	CUAV (v5)
1 GHz 32-bit single-core ARM1176JZF-S (0.319 GFLOPS)	Not applicable	Raspberry pi-zero
AM4x processor with ARM Cortex A9	Not applicable	Sitara (TI)
64-bit ARM Cortex A57 CPU, 4 cores (16 GFLOPS)	472 GFLOPS (e.g., NVIDIA GPU 128-core); variable power consumption 5 W to 10 W)	Jetson Nano, Jetson tx2
NVIDIA Carmel ARM® v8.2 64-bit CPU, 6 cores	21 TFLOPS (e.g., 384-core NVIDIA Volta GPU with 48 tensor cores)	Jetson Xavier NX
Intel Core i7/ i5/ i3, Celeron Mobile Processor	Integrated graphics	Mini-ITX, Intel NUC boards

3.3. COMPUTING ARCHITECTURES

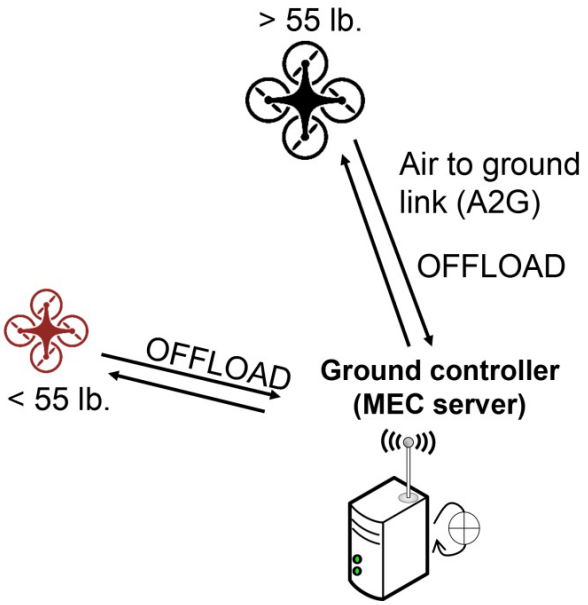
3.3.1. OVERVIEW

Most UAVs, especially the < 55 lb. type, do not have enough computing capability to process sophisticated algorithms: these UAVs can be thus referred to as computing-constrained, or reduced-function nodes. The traditional solution for such UAVs is to transmit (off-load) all the sensing data to a remote cloud edge server (ground controller/station) that can offer computing resources for processing. On the other hand, such a ground-based computing paradigm brings high latency, which cannot be ignored especially for some critical collision avoidance and latency-sensitive maneuvering tasks. Meanwhile, the expected increasing number of UAV computing tasks is expected to cause a large burden to the ground servers as well as privacy-leakage risks. To alleviate the bottlenecks at ground servers, UAVs are expected to fully utilize both ground and local (on-device) computing resources as a supplementary decision-making function. The sections that follow highlight some critical computing architectures often associated with UAV communications and related sensing tasks. For each architecture, critical computing constraints that UAV devices should satisfy are identified.

3.3.2. TERRESTRIAL EDGE COMPUTING

As depicted in FIGURE 2, the simplest computing scenario features UAVs acting as end users that need to execute computation tasks with the help of a terrestrial (edge) computing device. The ground controller is serving as an edge server, possibly mobile (i.e., a mobile edge controller [MEC]).

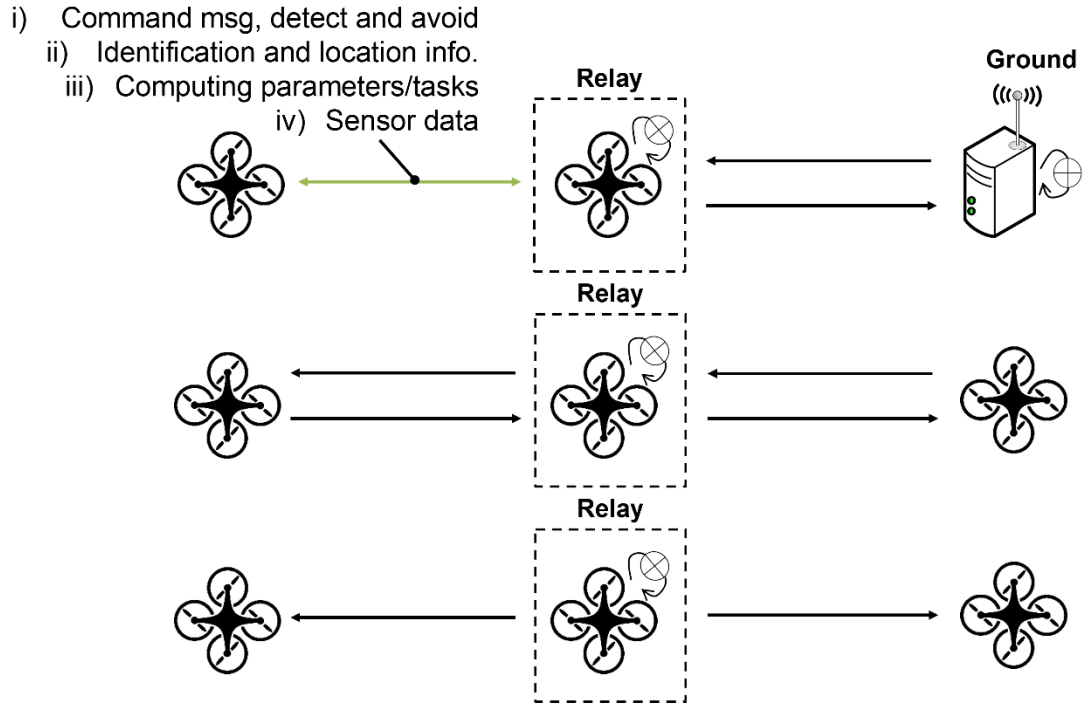
FIGURE 2 Terrestrial edge computing



In more detail:

- All UAVs offload all tasks to ground/terrestrial edge computing; no use of UAV-to-UAV communications.
- The architecture is suitable for < 55 lb. UAV with finite battery capacity and computing capabilities.
- Requirements on processing/computing power are low, estimated in the order of < 0.05 GFLOPS, that could be easily satisfied by state-of-art SoCs.
- Unicast messages only (no support for multicast/broadcast).

FIGURE 3 Example: UAV with relay functions (relay to UAV or terrestrial MEC)



3.3.3. RELAY FUNCTIONS

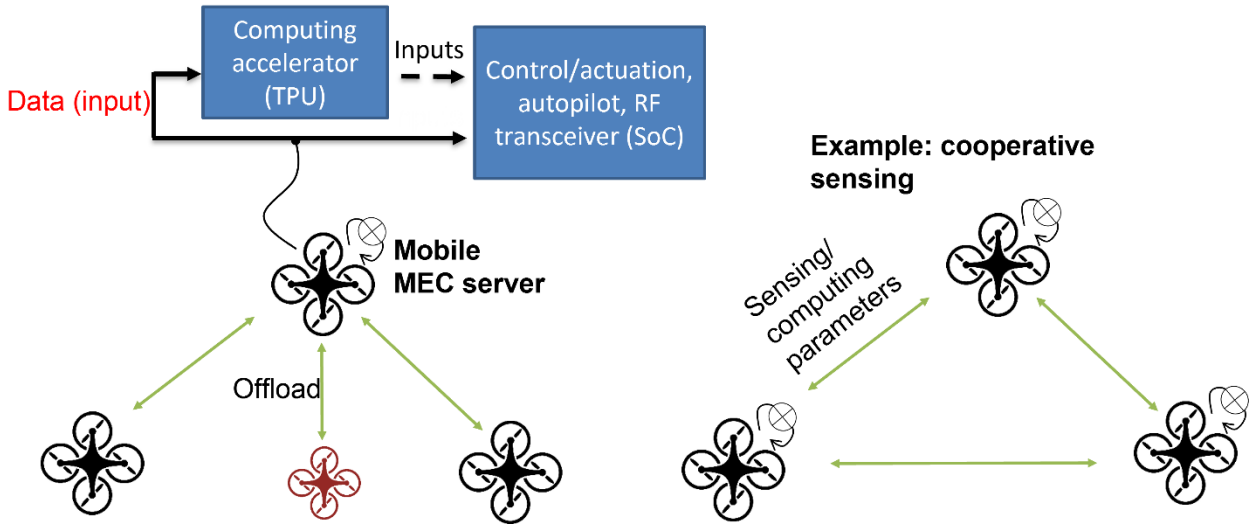
The UAV acts as a relay and assists other UAVs to either broadcast messages to neighbor UAVs (see 3.3.4) or offload their computation tasks to a ground station or terrestrial MEC server.

As depicted in FIGURE 3, this computing scenario could be generalized to multiple UAVs as well as UAV corridor applications, namely virtual highways in the sky for transportation, where communication and relaying functions are needed to maintain the minimum safety distance between vehicles. Other envisaged scenarios relate to ground station or MEC coverage extension (when poor wireless links are observed) using UAV-to-UAV communications.

The estimated processing power requirements are larger than the previous case as the UAV needs to support several relay functions as well as multicast, broadcast, and unicast UAV-to-UAV communication sessions. We estimate the minimum required processing power as approximately 0.05 GFLOPS for each UAV served. Note that collision avoidance applications may require more stringent processing power requirements.

Finally, memory requirements are also more stringent, preventing the use of simple SBC components (see examples in TABLE 4) that may not have enough cache size to store the relayed message payloads.

FIGURE 4 Example: UAV enabled mobile-edge computer



3.3.4. UAV ENABLED MOBILE SERVER

As depicted in the complex scenario of FIGURE 4, the UAV now acts as a MEC server and helps other UAVs (i.e., < 55 lb.) in proximity to compute specific tasks (e.g., trajectory replanning, deconfliction, and other sensor/data processing) on request (after the requesting UAVs offload their computational tasks). In particular, the usage scenario is set to replace terrestrial MEC server functions in the absence of a terrestrial MEC server or ground station.

Processing power requirements are generally higher than those required for relaying functions. Therefore, the UAV must now be equipped with a dedicated computing accelerator depending on the task (TPU/GPU, see 3.2). In more advanced scenarios, multiple UAVs might cooperate to execute the tasks (or sub-task). This requires the exchange of sensing data (group collision avoidance) or computing parameters over direct links.

4. UAV COMMUNICATION METHOD, MESSAGE FORMATS, AND USAGE

4.1. OVERVIEW

In this section the following will be discussed in greater detail:

- **UAV communication method**—Describes the communication methodology.
- **UAV message format**—Describes the UAV message formats.
- **UAV message prioritization based on craft type/usage**—UAV messages will be assigned up one of three levels of security and prioritization based upon the craft type and specific usage.
- **Remote ID**—All the message elements required in the Remote ID standard.
- **UAV obstacle tracking**—All UAVs are expected to track known obstacles detected within their immediate sensor operating range.

4.2. UAV COMMUNICATION METHOD

4.2.1. OVERVIEW

Several different communication formats were considered, and it has been determined that the best method for communicating would utilize a type of mesh network similar to existing Wi-Fi mesh network standards. This will allow other UAVs to act as repeaters in situations where the UAV originating the communication has either limited broadcast range due to the surrounding environmental conditions or technical limitations. Due to the expected density of communications in certain areas, the working group proposed using a limited flood mesh network. The communication network will use existing security protocols for establishing communication sessions between UAVs and will incorporate authentication and encryption techniques to secure communications.

There will be three types of communications being achieved between UAVs: broadcast messages, unicast messages, and multicast messages. An example of a broadcast message is the FAA's Remote ID requirement. To meet this requirement, all UAVs will broadcast their unique ID and location information to comply with the FAA Remote ID requirement. However, it is deemed that all broadcast messages should be limited to being repeated by other UAVs only if they are within one kilometer range of the UAV originating the broadcast message. This should help reduce communication traffic, keeping network traffic to a minimum so that the network does not get overwhelmed. There will be no such restrictions on unicast messages or multicast messages (queries, responses, or payload deliveries) routed to dedicated recipients. These unicast and multicast messages will continue to be propagated within the network until they reach the intended recipient.

4.2.2. UAV IDENTIFICATION

In similar fashion to the Remote ID standard, all UAVs will use some type of personalized ID to denote their identity. It will be included in all broadcast and query messages by the UAV initiating the message. Any UAV responding to a query message with a reply will likewise include its unique ID as part of the message. The UAV ID can be something permanently linked to the UAV, such as a manufacturer's serial number or something more temporary, like the session ID mentioned in the Remote ID regulation.

4.3. UAV MESSAGE FORMATS

Several different message formats were determined as necessary. They are broken up into five main categories: REMOTE_ID, DETECT_AVOID, NAV_COMD, DATA_PAYLOAD, and EMERGENCY_BROADCAST. The message formats will be explained in greater detail, starting with TABLE 5.

TABLE 5 Message formats sent/received by UAVs

Security requirement	Priority level	Message code	Value	Type	Data
		UNUSED	0x00		
Authenticated only	High	REMOTE_ID	0x01	Broadcast	Identification and location information
Authenticated and encrypted	Medium	EXTD_DETECT_AVOID_RESQ	0x02	Unicast/multicast	Request extended obstacles info GIS file format
Authenticated and encrypted	Medium	DETECT_AVOID_RESP	0x03	Unicast/multicast	Response/ACK to avoid-and-detect message
Authenticated and encrypted	High	NAV_COMD	0x04	Unicast	Navigation command message
Authenticated and encrypted	High	NAV_COMD_RESP	0x05	Unicast	Response/ACK navigation command message
Authenticated and encrypted	Low	DATA_PAYLOAD_RESQ	0x06	Unicast	Request for data payload delivery
Authenticated and encrypted	Low	DATA_PAYLOAD_RESP	0x9	Unicast	Response/ACK for Data Payload delivery
Authenticated and encrypted	Low	DATA_PAYLOAD	0x10	Unicast/multicast	Delivery of payload data to multiple UAVs
Authenticated only	High	EMERGENCY_BROADCAST	0x11	Broadcast	Describe nature of emergency
		RESERVED	....		

NOTE—To realize security objectives, the vision for a V2V datalink system relies on various components of a cybersecure environment (e.g., the use of cryptographic keys for authentication and confidentiality) (General Aviation Manufacturers Association [3], 5.1 Message Set and Protocols).

- REMOTE_ID

  REMOTE_ID has the broadcast message in compliance with the FAA's Remote ID regulation. It would allow receiving craft to determine position information of craft flying without transponders.

- DETECT_AVOID

  The DETECT_AVOID classification message will provide UAVs crowdsourced obstacle data from other craft and update its flight path to avoid possible collisions with unforeseen obstacles. It has a Query message that can be used to extract a compiled database of obstacles (static or moving) that the craft being queried has detected with its own sensors or has been informed about by other UAVs. The Response message allows the queried craft to respond to queries with acknowledgements or requested information.

- NAV_COMD

  The NAV_COMD classification is intended to be used for updating flight path or mission requirements for a UAV. It only uses unicast messages. NAV_COMD is used to address a specific UAV and provide updated navigation commands from the command-and-control station. The UAV can use NAV_COMD_RESP to acknowledge or respond to any received command message.

  An example scenario would be where the command-and-control station has received updated forecasting of hurricane activity in the flight path of a UAV and decided to re-direct the UAV to avoid the hurricane using NAV_COMD messages. The UAV can respond to acknowledge that it has received the updated navigation command that re-routes it around the hurricane.

- DATA_PAYLOAD

  The DATA_PAYLOAD classification uses both unicast and multicast messages. DATA_PAYLOAD_RESQ is used to query a UAV for its data payload information. DATA_PAYLOAD_RESP is used to acknowledge or respond to any data payload queries it receives. DATA_PAYLOAD can be sent either unicast or multicast, as appropriate, to whichever UAV(s) requested it.

  An example scenario using the DATA_PAYLOAD message could be crop surveillance when a farmer is using a swarm of 10 UAVs to find out how many acres of his crops were damaged by recent floods. Data from individual UAVs could be shared between all the UAVs to develop a composite image of the inspected crops.

- EMERGENCY_BROADCAST

  The EMERGENCY_BROADCAST classification was intended to allow a UAV to provide additional information that would better describe any emergency situations.

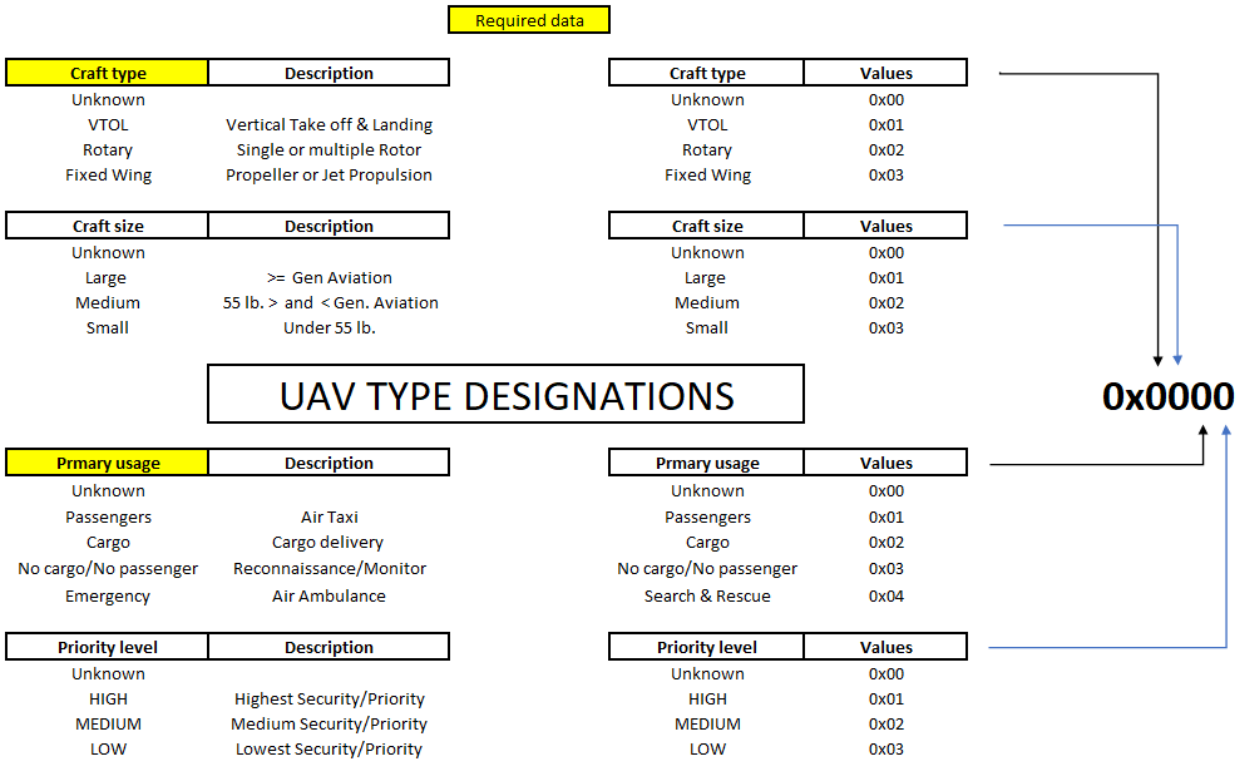
  An example scenario would be several UAVs patrolling a chemical processing plant that was recently evacuated after an explosion. They could send out an emergency message alerting other vehicles if any hazardous chemicals had subsequently leaked and been detected by any of the UAVs.

4.4. UAV MESSAGE PRIORITIZATION BASED ON CRAFT TYPE/USAGE

Conventional network protocols will be utilized to ensure secure communication using authentication and encryption. Higher levels of encryption may be used based on the computation capabilities of the UAVs involved. The different levels of craft type coupled with their primary usage will be used to determine the prioritization of the UAV message relative to other network traffic and therefore are the minimum required data to formulate the UAV type designation that will be embedded into each message. The UAV type designation will then be utilized to determine the prioritization of messages through the network.

The type designation can also be used in high density traffic situations where multiple UAVs are traveling through the same airspace to organize and prioritize traffic flow. Deference would be provided to emergency vehicles that are given the highest priority just like emergency vehicles traveling on roads and highways are given preferential treatment and priority. This ranking by craft type and usage could be useful in helping to alleviate UAV traffic congestion by knowing the movement capabilities and usage of UAVs in congested areas.

FIGURE 5 UAV type designations based on craft type and primary usage



4.5. REMOTE ID

The FAA has already issued a rule documenting what is required for transmitting Remote ID broadcasts.

FIGURE 6 FAA Remote ID minimum message elements

§ 89.305 Minimum message elements broadcast by standard remote identification unmanned aircraft.

A standard remote identification unmanned aircraft must be capable of broadcasting the following remote identification message elements:

(a) The identity of the unmanned aircraft, consisting of:

(1) A serial number assigned to the unmanned aircraft by the person responsible for the production of the standard remote identification unmanned aircraft; or

(2) A session ID.

(b) An indication of the latitude and longitude of the control station.

(c) An indication of the geometric altitude of the control station.

(d) An indication of the latitude and longitude of the unmanned aircraft.

(e) An indication of the geometric altitude of the unmanned aircraft.

(f) An indication of the velocity of the unmanned aircraft.

(g) A time mark identifying the Coordinated Universal Time (UTC) time of applicability of a position source output.

(h) An indication of the emergency status of the unmanned aircraft.

The Remote ID minimum message elements are defined in TABLE 6.

TABLE 6 Defined Remote ID message elements

					Units
Craft ID	=	Manufacturer's serial number	=	xxxxxxxxxxxxxxxxxxx	Letters and numbers
		Session ID	=	0x0000000000	Hexadecimal
Control station	=	Longitude	=	000 deg 00' 00.0000'' E or W	Degrees, sec, min
		Latitude	=	00 deg 00' 00.0000'' N or S	Degrees, sec, min
		Altitude	=	00000.00	Meters
Unmanned craft	=	Longitude	=	000 deg 00' 00.0000'' E or W	Degrees, sec, min
		Latitude	=	00 deg 00' 00.0000'' N or S	Degrees, sec, min
		Altitude	=	00000.00	Meters
		Velocity	=	Vx + Vy + Vz	Meters/Hour
		Status	=	0x0	Hexadecimal
UTC time mark	=	Time	=	00:00:00	HH:MM:SS
Manufacturer's serial number (ANSI/CTA-2063-A [1]) =	[4 character MFR CODE][1 character LENGTH CODE][15 character MANUFACTURER'S SERIAL NUMBER]				
Longitude =	(E)ast = positive and (W)est = negative				
Latitude =	(N)orth = positive and (S)outh = negative				
Status 0x1 =	Normal				
Status 0x3 =	Emergency (Non-normal)				
0000.00	= Vx				
0000.00	= Vy				
0000.00	= Vz				

4.6. UAV OBSTACLE TRACKING

The UAVs will be using their own sensor suites to detect obstacles while traveling along their flight path. It is expected that the UAV will be tracking these detected obstacles in a database stored in onboard memory. While traveling, the UAV will also be communicating with other UAVs and can query them for their database of detected obstacles. This would be done via an EXTD_DETECT_AVOID_RESQ query message. The UAV can then use this crowdsourced list of obstacles to do validation and verification of the obstacles it detected with its own sensor suite. Additionally, it can use this additional database list of obstacles to pre-emptively make course adjustments to its flight path to avoid obstacles it hasn't detected but have been detected by other UAVs. This would be similar to GAMA's local hazard warning for things such as icing or turbulence (General Aviation Manufacturers Association [3], Section 4.6 Local Hazzard Warning).

For example, a UAV has a flight path that takes it though a steep mountain canyon to deliver a package. However, due to a change in the wind, an uncontrolled forest fire has spread to the northern end of the canyon, engulfing it in flames. The UAV cannot detect the fire on the northern end of the canyon as it enters the canyon. It encounters some UAVs in use by firefighters to monitor the fire's progression through the area with specialized thermal sensors. The UAV delivering the package is able to communicate with the firefighters' UAVs and query them for a listing of obstacles. They respond with obstacle data that shows that the UAV's final destination (the customer's home) is currently on fire. Therefore, the UAV can make the determination that the delivery should be aborted and return to base since there is high probability, based on this new obstacle data, that the cargo and/or UAV would be destroyed by the fire.

The use of obstacle data from multiple other UAVs would improve the accuracy of detecting obstacles. This would be particularly useful for detection and tracking of more dynamic obstacles, such as storms or large flocks of birds, that would be more fluid over time. Additionally, it could effectively extend the range of detected obstacles with better accuracy of obstacle data as more UAVs' data are compiled.

5. REFERENCES

The following sources either have been referenced within this paper or may be useful for additional reading:

[1] ANSI/CTA-2063-A, Small Unmanned Aerial Systems Serial Numbers, 16 Sep. 2019.

[2] Code of Federal Regulations Title 14 Part 89, Remote Identification of Unmanned Aircraft.

[3] General Aviation Manufacturers Association (GAMA). "Vehicle-to-Vehicle Datalink Communications: Enabling Highly Automated Aircraft and High-Density Operations in the National Airspace." GAMA EPIC Data Communications Ad-hoc Committee Concept Paper, Version 1.0, 17 Dec. 2021.

RAISING THE WORLD'S STANDARDS

3 Park Avenue, New York, NY 10016-5997 USA http://standards.ieee.org

Tel.+1732-981-0060 Fax+1732-562-1571