

The Applicability of ROS in the Use of Unmanned Combat Aerial Vehicles

János Sári
Alba Regia Technical Faculty
Óbuda University
Székesfehérvár, Hungary
sari.janos1999@gmail.com
ORCID: 0000-0001-8861-3300

Károly Széll
Alba Regia Technical Faculty
Óbuda University
Székesfehérvár, Hungary
szell.karoly@amk.uni-obuda.hu
ORCID: 0000-0001-7499-5643

Abstract— Thanks to the technological advancement in the past few years the Unmanned Aerial Vehicles (UAVs) have developed immensely. It is inevitable to use UAVs for surveillance, reconnaissance and intelligence purposes in modern warfare. As a result of their high efficiency using more than one UAV in a mission has become incredibly widespread. Of course, drones required not only professionals on the field but also experts in the lab. Numerous algorithms are needed to enhance the UAV's intelligence in guidance, navigation, and control. One of the possibilities to upgrade drones in order to fulfil the requirements is the Robot Operating System (ROS). This paper introduces the possibilities of ROS based on its capabilities.

Keywords— UAV, ROS, framework, military

I. INTRODUCTION

Unmanned Aerial Vehicles (UAVs) can be found in many industries and environments due to their versatility. Of course, for UAVs to operate properly, it is essential to use and operate devices that can meet the different criteria. The resulting complex system is known as UAS (Unmanned Aircraft System) which has adopted by the Federal Aviation Administration (FAA). UAS include not only the Ground Control Station (GCS) and the Unmanned Aircraft but also the Command and Control Link/Data Link. In 1998 NATO started a standardization process in order to unify the unmanned control systems and help enable UAS interoperability. As a result, STANAG 4568 has been created which clarifies the definition of data link interface, control interface and Human and Computer Interfaces within NATO (see Fig. 1.).[1][2][4]

In modern warfare using UAVs are not only inevitable but also essential in order to aid soldiers and protect human life. Drones can be controlled remotely or can fly autonomously (based on a pre-programmed flight plan); also, they are capable of operating individually and in swarms. Nowadays it can be risky to use one drone for a specified mission because of the new threats that have emerged in the recent years (EMP attacks, hijacking, Data link spoofing, jamming) [16]. Due to this, research in the field of swarm intelligence and artificial intelligence is becoming increasingly important.[9]

One way to enhance UAS while maintaining the efficient development is using software like ROS (Robot Operating Systems). ROS not only provides hardware abstraction and low-level device control but also message-passing between processes for system.[7]

In this paper the integrability of ROS in UAS is investigated. After a brief description about the possible

ways of grouping drones and the UAVs used in military, the basics of ROS will be discussed connected to the topic. The following sections describe the vulnerabilities of Robot Operating Systems, the problems that might arise, the solutions, and the basic requirements of a ROS based UAV.

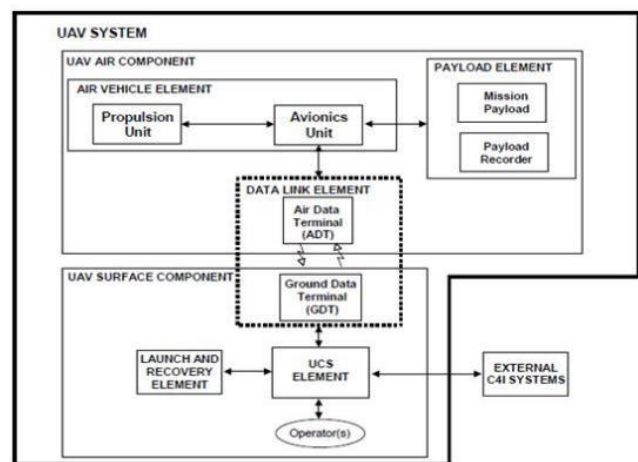


Fig. 1: Possible UAS Architecture (STANAG 4586) [3]

II. CLASSIFICATION OF UAVs

The classification of the Unmanned Aerial Vehicles can be done in several ways due to their universal use and diversity. There are differences in size (micro), design (VTOL, tilt-wing, HTOL) and also in the incorporated energy source (battery, fuel cells). Of course, as technology continues to evolve, new types will emerge, sometimes forming entirely new groups (see Fig. 2.). [6][8]

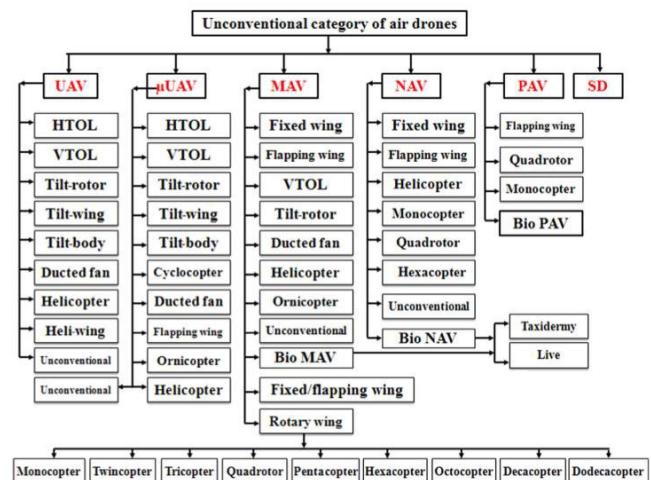


Fig. 2: Various types of UAVs [8]

III. UAVs FOR MILITARY PURPOSES

As we mentioned UAVs can be used for different kind of military purposes: surveillance, reconnaissance, intelligence etc. They can also intervene on the battlefield either indirectly using precision guided munitions controlled by an operator or directly by dropping or firing these munitions autonomously.[14][15]

The history of the first military Unmanned Aerial Vehicle dates back to the Second World War. The result of the competition between France and the US was the Voisin BN3 biplane which was able to fly for about 100km. However, UAVs were not used for surveillance purposes until 1973, during the Vietnam War.[14][15]

Although a classification of unmanned aerial vehicles has been presented above, a classification as function is preferred for military use. On this basis we can distinguish:

- Target and Decoy UAVs
- Reconnaissance UAVs
- Combat UAVs
- Research and Development UAVs
- Civil and Commercial UAVs

The different classifications clearly define the difficulty or the risks of the mission and its purpose. Of course as technology evolves, these groups are also expanding. One such application – that has been explored by the authors as well – is the deployment and use of various UAVs to enhance the security and runway safety of military airfields. [14][15]

Using drones with thermal imaging camera can improve the accuracy of weather mapping. With the image analysis of images along the runway UAVs can help to detect the presence of foreign object debris (FOD) on the surface (runway), such as fallen parts of the aircraft or a flock of birds. Furthermore, changes in the condition of surfaces, cracks and icing can be detected (see Fig. 3).[9]

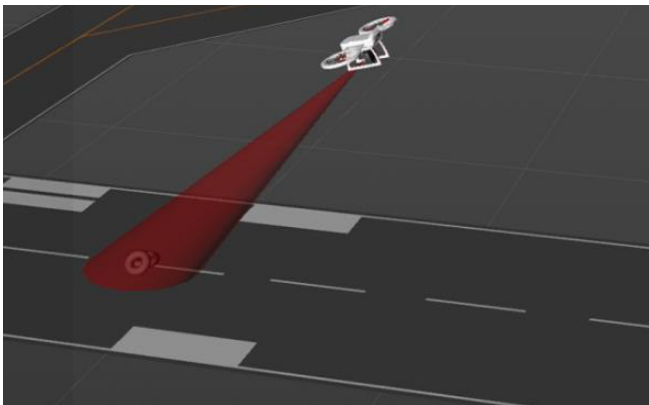


Fig. 3: Use of unmanned aerial vehicle as a FOD seeker

IV. ROBOT OPERATING SYSTEM (ROS)

The Robot Operating System, or ROS, is a framework widely used in robotics. It is an open source; American-developed; meta-operating system that uses libraries to build, write and run codes across multiple computers. It has the advantage of having many pre-built features such as face recognition, motion tracking, robot control, route planning etc. ROS is mainly based on Linux, C++ and Python

programming languages. The applications created in ROS are called nodes (see Fig. 4.) (master, publisher, and subscriber). The communication between nodes is implemented by the ROS system itself. [1][7]



Fig. 4: ROS structure

In case of asynchronous communication in ROS, the preferred model is the publisher-subscriber model. The advantages of this model are:

- Any code can publish a message to any topic.
- Any node can subscribe to any topic.
- Multiple nodes can publish to the same topic.
- Multiple nodes can subscribe to the same topic.
- A node can publish to multiple topics.
- A node can subscribe to multiple topics.

In order to support different kind of programming languages ROS include a client library. Of course for other purposes for example hardware control, transmission and reception, visual space controlling; ROS has its own relevant component: hardware interface, communication, simulation tools (see Fig. 5.).



Fig. 5: Components of ROS [7]

As we mentioned ROS is an open source, meta-operating system which allows us to use it in the field of unmanned aviation. In a structure where ROS is implemented into an UAV or UAS, the aforementioned pub-sub model is usually adopted for communication. In this case the master node helps in the information exchange process between the two other nodes. [1]

To make the communication process easier between small drones a protocol so called MAVlink is used. It follows a modern hybrid publish-subscribe and point-to-point design pattern. Thanks to the diversity of ROS, the possibilities offered by MAVlink can be exploited by MAVROS (see Fig. 6.). In fact MAVROS is a MAVlink expandable communication node. [5]

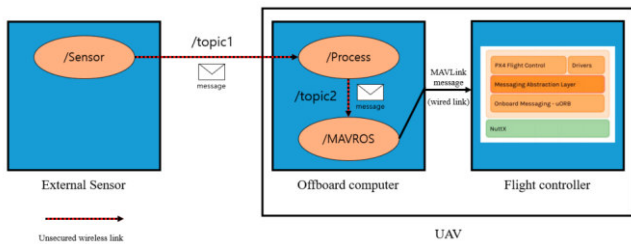


Fig. 6: A possible structure of a ROS-based UAV [1]

As we mentioned there are many disadvantages using ROS in UAVs. The weak points are the possibility of jamming, Data link spoofing and in extreme cases hijacking [16]. Of course software developers are aware of the risk factors. Due to this users are advised to create a restricted access to the network, and not disclosing the ROS master. It can be achieved by establishing isolated hosts, networks, or using firewalls. Additional problems can be caused by the formation of abnormal or infinite loops. For this reason a timer tool called Watchdog is implemented in ROS which monitors the CPU and intervene when loops may occur. [1][7]

Because of the vulnerability of the systems it is easy to break into the network. Attacks like false data injection can easily lead to total collapse. Unfortunately the weaknesses of the master node (for example the master node does not check whether the node making the request is a normal node or a malicious node) are the main reason why this can happen. [1]

V. HOW TO SECURE ROS

Apart from the mentioned solutions there are many other options for securing our ROS system. In the following we would like to present two alternatives that have become quite popular in the recent days.

Encryption

Many studies show that ROS-Encryption is not an idea to be rejected. Ciphered communications in ROS requires just a little more overhead of CPU performance but it provides a much more secure flow of information.

The number of algorithms which can be used for encryption has increased significantly in the past years. Nowadays cryptographic algorithms AES (Advanced Encryption Standard) are preferred over DES (Data Encryption Standard) due to their speed and reliability. Algorithms like AES converts data in plaintext to an uninterpretable form which is called the ciphertext.[20]

Secure-ROS

Secure-ROS provides alternative versions of core ROS packages that enable secure communication between ROS nodes. Secure-ROS uses IPsec (Internet Protocol Security) in transport mode and modified versions of the ROS master (rospy, roscpp) to ensure secure communication. The user only can specify authorized subscribers and publishers to topics at run time.[19]

VI. THE BASIC REQUIREMENTS FOR ROS BASED UAVS

The first and most important task is to check whether the integration of the drone into ROS is even possible. This step is crucially important because complete ROS systems can be very expensive, and due to this they are not always cost-

efficient. Thankfully, the mentioned problem can be solved easily with ROS plugins. An additional solution is the use of ROSBRIDGE, which provides a JSON API to ROS functionality for non-ROS programs (the client does not have to install ROS).[12][13]

The second step is the correct choice of the frame and the propeller. The size, structure and the material of the frame have a significant influence on the flight characteristics. The properties of the propeller will mainly affect the lifting power. To study the behavior of the propeller and the frame, it is worth creating a simulation environment. This can be done with software like Ansys (see Fig. 7.) which is a numerical fluid dynamics software that discretises continuous operations in space and time. [11]

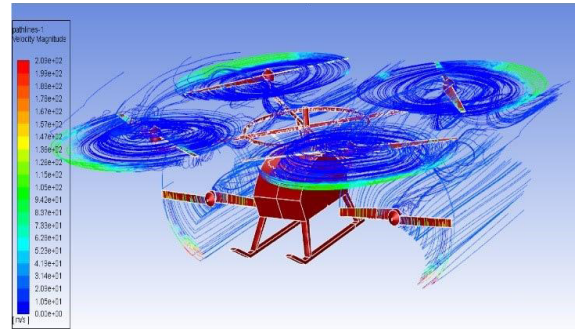


Fig. 7: A possibility of a UAV Ansys simulation [10]

As already mentioned, drones or UAVs can also be distinguished by the energy source they use. For small unmanned aircrafts, as shown in Figure 3, lithium polymer batteries are the preferred choice. Compared to lithium-ion, these types are more resistant to overcharging and less likely to leak electrolyte. Another feature is that they are much lighter than their predecessors, thanks to the special polymer.

A further necessary condition is the existence of engines. The number and the power of engines are mainly affected by the goal which we want to achieve. Of course, it is not enough to install the engines themselves, they also need to be synchronized. In case of small drones the brushless engines are more common due to their durability and performance.

Other essential components of drones are the various control elements such as Electronic Speed Controller, Flight Controller, Manual Control Unit etc. (see Fig. 8.) In the case of embedded system, the applications of fault-tolerant redundant solutions are recommended [17], [18].



Fig. 8: Example of drone main parts

VII. CONCLUSION

The experience of recent years suggests that unmanned aircraft will become more widespread in the coming years. Of course, this will influence many civilian and military areas as well. UAS will become more complicated, but hopefully the communication between drones becomes easier. But of course this requires a system that can effectively manage the parties involved in the communication process. According to the current state of technology, one such solution could be ROS.

The use of ROS in military drones is not always advantageous, as can be seen above, but it offers a very practical solution thanks to its ease of use and simplicity. It is easy to see that systems such as ROS would be able to innovate, this branch of warfare, but we assume that ROS itself will only be used by underdeveloped forces.

This study not only presents the classification of drones and their system architecture, but also takes a new look at the application of unmanned aerial vehicles using ROS.

ACKNOWLEDGMENT

The authors thankfully acknowledge the financial support of this work by the Howmet Aerospace Foundation, the Arconic Foundation, and the project no. 2019-1.3.1-KK-2019-00007 implemented with the support provided from the National Research, Development and Innovation Fund of Hungary, financed under the 2019-1.3.1-KK funding scheme.

REFERENCES

- [1] Hyojun Lee, Jiyoung Yoon, Min-Seong Jang, Kyung-Joon Park: A Robot Operating System Framework for Secure UAV Communications, *Sensors* 2021, 21, 1369. [CrossRef]
- [2] Suraj G. Gupta, Mangesh M. Ghonge, Dr. P. M. Jawandhiya: Review of Unmanned Aircraft System (UAS), *International Journal of Advanced Research in Computer Engineering & Technology (IJARCET)* Volume 2, Issue 4, April 2013
- [3] Michael Chanda, Julee DiPlacido, John Dougherty, Richard Egan, John Kelly, Trent Kingery, Daniel Liston, Douglas Mousseau, James Nadeau, Theodore Rothman, Lisa Smith, Michael Supko: Proposed Functional Architecture and Associated Benefits Analysis of a Common Ground Control Station for Unmanned Aircraft Systems, *NAVAL POSTGRADUATE SCHOOL MONTEREY, CALIFORNIA*, Available online: <https://www.researchgate.net/publication/265528436> (accessed on 25.10.2021)
- [4] Mário Monteiro Marques: STANAG 4586 –Standard Interfaces of UAV Control System (UCS) for NATO UAV Interoperability, *STO-EN-SCI-271*, Available online: <https://www.sto.nato.int/publications/STO%20Educational%20Notes/STO-EN-SCI-271/EN-SCI-271-03.pdf> (accessed on 25.10.2021)
- [5] MAVlink: MAVLink Developer Guide, Available online: <https://mavlink.io/en/> (accessed on 25.10.2021)
- [6] Adnan S. Saeed, Ahmad Bani Younes, Chenxiao Cai, Guowei Cai: A survey of hybrid Unmanned Aerial Vehicles, [CrossRef]
- [7] YoonSeok Pyo, HanCheol Cho, RyuWoon Jung, TaeHoon Lim: ROS, Robot Programming, From the basic concept to practical programming and robot application, Dec 22, 2017, ISBN: 979-11-962307-1-5
- [8] M. Hassanalani, A. Abdelkefi: Classifications, applications, and design challenges of drones: A review, [CrossRef]
- [9] Vas Tímea, Békési Bertold, Sári János, Kele Katalin, Magyar Martin, Szabó Richárd: A mesterséges intelligencia alkalmazhatósága a modern kori repülésben, *Repüléstudományi Közlemények*, 33. évfolyam (2021) 1. szám 5–18., DOI: 10.32560/rk.2021.1.1
- [10] Ansys simulation. Available online: <https://www.fiverr.com/cfd828/do-cfd-simulation-ud-with-ansys-fluent> (accessed on 25.10.2021)
- [11] Sári János, Beneda Károly, Kavas László: Repülőgép-hajtóművek égésterseinek áramlástan vizsgálatára számítógépes szimulációban, *REPÜLÉSTUDOMÁNYI KÖZLEMÉNYEK (1997-TŐL)* 32 : 3 pp. 93-103. , 11 p. (2020) [CrossRef]
- [12] Rosbridge. Available online: http://wiki.ros.org/rosbridge_suite (accessed on 25.10.2021)
- [13] Pluginlib. Available online: <http://wiki.ros.org/pluginlib> (accessed on 25.10.2021)
- [14] John W.R. Taylor, John F. Guilmarin: Unmanned aerial vehicles (UAVs). Available online: <https://www.britannica.com/technology/military-aircraft/Unmanned-aerial-vehicles-UAVs> (accessed on 25.10.2021)
- [15] Jack Brown: TYPES OF MILITARY DRONES: THE BEST TECHNOLOGY AVAILABLE TODAY Available online: <https://www.mydronelab.com/blog/types-of-military-drones.html> (accessed on 25.10.2021)
- [16] Judit Takács MÓDNÉ: The importance and development of safety awareness with soft skills in industrial environment, *OPUS ET EDUCATIO: MUNKA ÉS NEVELÉS* 8 : 3 pp. 314-323. , 10 p. (2021)
- [17] Györök György, Beszédes Bertalan. Fault-tolerant Software Solutions in Microcontroller Based Systems. In: Orosz, Gábor Tamás (szerk.) AIS 2017 - 12th International Symposium on Applied Informatics and Related Areas organized in the frame of Hungarian Science Festival 2017 by Óbuda University : Proceedings. Székesfehérvár, Magyarország : Óbudai Egyetem (2017) 204 p. pp. 7-12. , 6 p.
- [18] György Györök, Bertalan Beszédes. Adaptive optocoupler degradation compensation in isolated feedback loops. In: Szakál, Anikó (szerk.) IEEE 12th International Symposium on Applied Computational Intelligence and Informatics (SACI 2018). Temesvár, Románia : IEEE Romania Section, IEEE Hungary Section (2018) 604 p. pp. 167-172. , 6 p.
- [19] Secure ROS. Available online: https://sri-csl.github.io/secure_ros/ (accessed on 25.10.2021)
- [20] Doczi, R.; Kis, F.; Suto, B.; Poser, V.; Kronreif, G.; Josvai, E.; Kozlovsky, M. Increasing ROS 1.x Communication Security for Medical Surgery Robot. In Proceedings of the 2016 IEEE International Conference on Systems, Man, and Cybernetics (SMC), Budapest, Hungary, 9–12 October 2016