SEARCH AND RESCUE ROBOTICS - FROM THEORY TO PRACTICE

Contributors

Geert De Cubber, Daniela Doroftei, Konrad Rudin, Karsten Berns, Anibal Matos, Daniel Serrano, Jose Manuel Sanchez, Shashank Govindaraj, Janusz Bedkowski, Rui Roda, Eduardo Silva, Stephane Ourevitch, Rene Wagemans, Victor Lobo, Guerreiro Cardoso, Keshav Chintamani, Jeremi Gancet, Pascal Strupler, Atabak Nezhadfard, Massimo Tosa, Haris Balta, Jose Almeida, Alfredo Martins, Hugo Ferreira, Bruno Ferreira, Jose Alves, Andre Dias, Stefano Fioravanti, Daniele Bertin, German Moreno, Jose Cordero, Mario Monteiro Marques, Alberto Grati, Hafeez M. Chaudhary, Bart Sheers, Yudani Riobo, Pierre Letier, Mario Nunez Jimenez, Miguel Angel Esbri, Pawel Musialik, Irune Badiola, Ricardo Goncalves, Antonio Coelho, Thomas Pfister, Karol Majek, Michal Pelka, Andrzej Maslowski, Ricardo Baptista

Notice

Statements and opinions expressed in the chapters are these of the individual contributors and not necessarily those of the editors or publisher. No responsibility is accepted for the accuracy of information contained in the published chapters. The publisher assumes no responsibility for any damage or injury to persons or property arising out of the use of any materials, instructions, methods or ideas contained in the book.

Contents

Introduction to the Use of Robotic Tools for Search and Rescue

Geert De Cubber, Daniela Doroftei, Konrad Rudin,
Karsten Berns, Anibal Matos, Daniel Serrano,
Jose Sanchez, Shashank Govindaraj,
Janusz Bedkowski, Rui Roda, Eduardo Silva and
Stephane Ourevitch

Additional information is available at the end of the chapter

Abstract

Modern search and rescue workers are equipped with a powerful toolkit to address natural and man-made disasters. This introductory chapter explains how a new tool can be added to this toolkit: robots. The use of robotic assets in search and rescue operations is explained and an overview is given of the worldwide efforts to incorporate robotic tools in search and rescue operations. Furthermore, the European Union ICARUS project on this subject is introduced. The ICARUS project proposes to equip first responders with a comprehensive and integrated set of unmanned search and rescue tools, to increase the situational awareness of human crisis managers, such that more work can be done in a shorter amount of time. The ICARUS tools consist of assistive unmanned air, ground, and sea vehicles, equipped with victim-detection sensors. The unmanned vehicles collaborate as a coordinated team, communicating via ad hoc cognitive radio networking. To ensure optimal human-robot collaboration, these tools are seamlessly integrated into the command and control equipment of the human crisis managers and a set of training and support tools is provided to them to learn to use the ICARUS system.

Keywords: robotics, search and rescue, crisis management, disaster management

1. Introduction: Why do we need search and rescue robots?

Recent dramatic events such as the earthquakes in Nepal and Tohoku, typhoon Haiyan or the many floods in Europe have shown that local civil authorities and emergency services

have difficulties in adequately managing crises. The result is that these crises lead to major disruption of the whole local society. On top of the cost in human lives, these crises also result in financial consequences, which are often extremely difficult to overcome by the affected countries.

In the event of large crises, a primordial task of the fire and rescue services is the search for human survivors on the incident site. This is a complex and dangerous task, which—too often—leads to loss of lives among the human crisis managers themselves. The introduction of unmanned search and rescue (SAR) devices can offer a valuable tool to save human lives and to speed up the search and rescue process.

Indeed, more and more robotic tools are now leaving the protected lab environment and are being deployed and integrated in the everyday life of citizens. Notable examples are automated production plants in industry, but also the widespread use of consumer drones and the rise of autonomous cars in public space. Also in the world of search and rescue, these robotic tools can play a valuable role.

Of course, this does not mean that the introduction of robotic tools in the world of search and rescue is straightforward. On the contrary, the search and rescue context is extremely technology unfriendly, as robust solutions are required which can be deployed extremely quickly. Chapter 2 of the book will give a more in-depth review of the requirements for search and rescue robotics, as proposed by the human users of these systems. Indeed, one crucial aspect must not be forgotten: the robotic tools must not have the objective to eliminate the need of human search and rescue workers! Instead, these robotic assets must be seen as yet another tool in the ample toolkit of human search and rescue workers in order to allow them to do their job better, faster, and safer. In the following paragraphs, each of these statements is further developed.

1.1. Better

As stated before, robotic search and rescue tools are there to assist human rescue workers. One of their main strong points is that they can increase the situational awareness of the relief workers by giving them a better and higher quality view on the nature of the crisis. Indeed, robotic tools are able to give better insights by looking at disaster scenes from a point of view which is nearly impossible (or impractical or very unsafe) to obtain by humans. One example is the use of drones which can provide a quick birds-eye view of a disaster scene, which is crucial information for the planning of rescue operations. Another example is the use of underwater robots for mapping debris or searching for human remains under the water, which is an essential recovery operation after floods, tsunamis, or typhoons have damaged and blocked ports and waterways.

The miniaturization of sensing technology has led to the result that search and rescue robots can pack more and more sophisticated sensors (high-definition video cameras, thermal cameras, 2D and 3D laser range finders, sensors for measuring chemical, biological, and radiological contamination, …), allowing for precise and fast cartography. Undisturbed by

cloud cover, these robotic assets are therefore becoming a very good complimentary tool to space-based remote sensing, which remains essential to cover large areas. The introduction of these advanced sensors on unmanned search and rescue robots opens the possibility to perform damage assessment operations with these unmanned assets, thereby keeping the human operators safe. Nowadays, unmanned systems are capable of producing accurate three-dimensional (3D) maps of the environment, pinpointing objects of interest (human survivors, but also potential dangers like fire hazards or chemical spills) in these 3D models. Such maps provide highly valuable information to human search and rescue workers in the assessment phase, where they need to decide which buildings/structures to enter first. These 3D maps also help for cartography of the debris after the crisis, which can be of help to coordinate the recovery operations and the structured removal of debris. Advances in telecommunication technology now also make it possible to let remote experts (possibly at the other side of the world) analyze damage to structures, based on live high-quality data gathered by unmanned systems. Such remote expert analysis can be invaluable to assess the structural integrity of buildings or shipwrecks.

Unmanned assets equipped with powerful sensors have an important role to play as data gatherers during a crisis, not only to support the immediate relief operations. Indeed, in the aftermath of a crisis, often a legal battle entails between people suffering from damages, the authorities, and insurance companies. Accurate, time-stamped and geo-referenced data collected by unmanned systems during the crisis can serve as evidence to settle these disputes. An example of this happening in practice is the detection by ad drone of illegal man-made dyke breaches during the 2014 floods in Bosnia-Herzegovina [1] (more information: see chapter 10).

Using unmanned assets can also have more sense from an economical point of view. Indeed, typical search and rescue operations at land or sea happen via the deployment of manned rescue helicopters and/or patrol boats, both costing thousands of dollars an hour to operate. Unmanned assets can drastically bring this operational cost down and free up the manned assets for high-priority tasks.

1.2. Faster

In a search and rescue context, time is a critical parameter, as the chance of survival of victims decreases quickly. It is therefore essential to deploy all the search and rescue assets as quickly as possible. However, during a large crisis, it is often the case that traditional search and rescue assets (rescue helicopters, rescue boats, …) are extremely overloaded, e.g., for evacuating victims. The fast deployment of ubiquitously present unmanned rescue tools can greatly speed up the rescue operations.

A main benefit of mostly the aerial unmanned tools is that they enable human rescue workers to very quickly obtain a global overview of the situation and the dangers in the crisis area. The result is that the search and rescue workers can thus plan their operations faster, not having to wait until satellite imagery is available or a ground-based survey is performed.

1.3. Safer

An obvious advantage of using robotic systems in comparison to their manned counterparts is that the unmanned systems keep the human rescue workers out of harm. This is especially important in earthquake response scenarios, where search and rescue workers now still have to enter semi-demolished structurally unstable buildings to search for survivors, terrified by the possibility of aftershocks bringing the whole structure down. Indoor drones and ground robots are specifically suited for these tasks.

Also crisis where there is a chemical, biological, or radiological component pose a huge problem for human relief workers, as proven by the dramatic events in Fukushima where a tsunami caused a meltdown of three nuclear reactor cores, exposing the environment to nuclear radiation. In such circumstances, robotic assets can be the only tools to correctly deal with the crisis, without endangering more human lives.

At sea, it is currently the case that rescue operations need to be halted at night or when the sea gets too rough, because it would be too dangerous for the human search and rescue workers. Robotic assets certainly do have difficulties as well with rough environmental conditions like night-time operation, heavy wind, rain or rough sea state, but in a risk-assessment context, it would be logical to deploy these unmanned systems instead of manned assets for risky operations. Furthermore, unmanned rescue tools show great promise for operations of victim search at sea during the night because it is easier to detect humans in the water than during the day (due to the larger thermal gradient between the human and the water) and the limited number of operations at night.

2. Search and rescue robotics efforts around the world

2.1. Internationally

From an operational side, the international urban search and rescue (USAR) community is organized via the INSARAG network [2], which falls under the United Nations umbrella. INSARAG establishes minimum international standards for USAR teams and methodologies for international coordination crisis response scenarios, based on the INSARAG Guidelines [3]. Via the elaboration of these standards, INSARAG drives technological development. The use of unmanned assets for crisis management has been acknowledged by the INSARAG group [4] and is one of the discussion points for the elaboration of future collaboration and coordination standards, in order to allow multi-national teams working in the same crisis area to share data from their unmanned assets. The International Maritime Rescue Federation [5] is taking up a similar—be it less globally coordinated—role in the world of marine search and rescue.

Support to operational deployment of robotic tools for search and rescue is given by initiatives as UAViators [6] and the Roboticists Without Borders program [7], where the former focuses on the use of aerial robotic tools (unmanned aerial vehicles or UAVs or drones) and the latter considers the use of all kinds of robotic tools (including marine and ground robots).

The objectives of the UAViators intiative are [8] to establish standards for the responsible use of UAVs and provide up-to-date regulatory information; document lessons learned and best practices; provide hands-on UAV training; inform UAV deployments after disasters; and catalyze research and information sharing. When a disaster strikes, the UAViators crisis map [8] is updated and UAV rescue teams can announce their capabilities and deployment details. The deployed UAV teams can then post data collected by their unmanned assets on this website, such that remote users, acting as digital humanitarians [9], can analyze the data. This approach of trying to organize and structure the relief operations with UAVs has led to some good results in the past, as can be read in a report [10] by FSD, CartONG and the Zoi Environment Network on the use of drones in humanitarian crises. As part of that report, they have created 14 success stories of the use of UAVs in crisis response, many of them with the help of people from the UAViators network.

The Roboticists Without Borders program [7] is an initiative by the Center for Robot-Assisted Search and Rescue (CRASAR) at Texas A&M University. It aims to create pools of professionals in ground, aerial, or marine robots or emergency response who are trained in disaster response and how to work with incident management, what are the types of missions and best match of systems with the needed data, and have participated in high-fidelity exercises. More geared toward the professional robotics community than the UAViators initiative, the Roboticists Without Borders program aims to find the right matches between universities, industry, and private individuals in order to deploy the right robotic systems to a particular incident, while at the same time gaining deeper insights into the needs and requirements of the disaster response community. CRASAR director Robin Murphy, founder of the Roboticists Without Borders program, has written an excellent book [11] on the subject of disaster robotics which describes different successful real-life deployments of this initiative (and others), including the scientific progress in the field.

From a scientific point of view, the international research direction in the field of Safety, Security, and Rescue Robotics is driven by a specific technical committee on this subject domain, launched by the Robotics and Autonomous Systems Group of the Institute of Electrical and Electronics Engineers. This technical committee was founded shortly after the first robots were deployed to help with the search operations during the 9/11 World Trade Center collapse, leading to an accelerated adoption of robots for homeland security and public safety. The primary activity for the committee is to engage emergency responders, federal and local government agencies, and non-governmental organizations for training and acquisition guidance.

Prototypes of robotic tools for search and rescue, developed in different laboratories worldwide, compete since 2001 annually with one another in the RoboCup Rescue competition [12]. This event—which falls under the umbrella of the RoboCup annual international robotics competition [13]—was inspired by the Kobe earthquake and pits robots to compete to find victims in a simulated earthquake environment. The robots have to operate totally autonomously and can score points by detecting victims and hazards and by mapping the environment. The aim of the competition is to encourage the transfer of academic research into the disaster-rescue domain, and to encourage research in a socially significant real-world domain, by offering a publicly appealing challenge [12].

2.2. United States of America

From 2012 to 2015, the US Defense Advanced Research Projects Agency (DARPA) has tried to increase the research and take-up of disaster response robotics by organizing a competition [14] where semi-autonomous robots had to execute a number of tasks in urban search and rescue disaster response scenarios. In order to end up with modular and versatile systems, these tasks were chosen very diverse and based upon present-day tasks executed by human search and rescue workers. Examples of tasks were driving with a vehicle, opening a door and entering a building, locating and closing a valve, and climbing a ladder [15]. The definition of the tasks led to the widespread use of humanoid-like robots in this event. The qualification to the event was dominated by the SHAFT robot by Google, which later withdrew from the challenge due to the military origins of the event. The competition was eventually won [16] by the Korean KAIST team with their humanoid HUBO robot, which managed to complete all tasks.

The US National Institute of Standards and Technology (NIST) plays an important role in the development of standardized test methodologies for search and rescue robotics [17]. Evolved from standardized test methodologies helping (primarily military) contractors validate and compare explosive ordnance disposal robots, NIST has developed specific test methodologies and standardized procedures for qualitatively and quantitatively evaluating the performance of search and rescue robotics. These NIST standardized test methodologies apply mostly to smaller ground robots, but are now also being extended to aerial robots and larger systems. The existence of standardized validation methodologies for search and rescue robotics is essential not only for scientists and developers to accurately compare multiple novel developments, but also for procuring agencies to choose the right robotic assets according to their specific needs.

Arguably, the institution contributing most to the introduction of robotic tools in the world of search and rescue is the aforementioned the Center for Robot-Assisted Search and Rescue (CRASAR) of the Texas A&M University [7]. CRASAR has as an objective to improve the crisis response lifecycle, by the introduction of robotic tools in the process. CRASAR members were among the first to deploy robotic tools for disaster management during the 9/11 attacks in 2001 and have since been actively involved in more than 15 documented deployments of disaster robots throughout the world, ranging from land to sea and air robots [11]. Associated to CRASAR is the Texas A&M Engineering Extension Service Disaster City testing grounds, featuring a training facility where human operators can learn to work with disaster management robots and where these robotic assets can be validated and compared to one another (e.g., following the NIST standardized test methodologies).

2.3. Far East

Located in a very disaster-prone area, countries like Japan, Korea, China and ASEAN member states have invested many resources in the development of novel disaster management tools, including robotic tools. These robotic tools were also put to use after the 2011 Great Eastern Japan Earthquake in Tohoku, Japan, where robotic assets, both from Japan as from the USA, were deployed to help in the disaster management operations [11, 18]. Ground and aerial robots helped for monitoring and surveillance operations, whereas marine robots assisted with clearing the harbors.

Following up on this disaster, prof. Tadokoro of the Tohoku University organized during the 2015 UN World Conference on Disaster Risk Reduction in Sendai, Japan, a public forum on the Social Implementation of Disaster Robots and Systems [19]. During this event, lessons learnt from past deployments of disaster robotics tools were discussed and remaining bottlenecks were identified. One of the conclusions was that the present-day generation of robotic tools for disaster management still often lack robustness to operate in the tough environments encountered in crisis management. Therefore, the Japanese government started a Tough Robotics Challenge research and development project [20] in the framework of the Impact program. Looking into the future, the Japanese efforts toward the development of search and rescue robotics are going to be driven by the on-going need of the use of robotics for the clean-up and dismantling of the four reactors of the Daiichi nuclear power plant damaged in the Fukushima accident and by the prospect of the "Robot Olympics" which will be organized next to the Summer Olympics in Tokyo in 2020.

Reports of robotic search and rescue tools deployed in China less frequently reach international coverage, but there are some important successes to be reported. Already in 2013, the Chinese International Search and Rescue Team was supported by an unmanned aerial vehicle of the State Key Robotics Lab at Shenyang Institute of Automation to help with the relief operations after the Lushan earthquake [21]. As a very fine example of how novel technologies are brought from the lab directly into the field, the unmanned system performed real-time feature detection of disaster damage from live aerial video footage, thereby speeding up the classification of the damages on the terrain.

In the aftermath of the DARPA challenge, won by the Korean KAIST team as reported before, South Korea and the United States have agreed to start a joint research project [22] aimed at developing the next generation of robotics system for disaster environments.

2.4. Middle East and Russia

Confronted with a huge and often very inaccessible territory to cover by the emergency services, the Russian Federation is also investing in search and rescue robots. The focus in Russia is more on developing systems which are able to deal with extreme environments and environmental conditions. Examples are operation in Siberian and Arctic temperatures [23], mobility in swampy forests (taiga), polluted (nuclear) infrastructure, wide area search operations, etc. Compared to other countries in the world, research efforts are therefore more concentrated on developing larger, robust systems [24] with advanced mobility features and autonomous terrain traversability analysis capabilities and on validating these technologies on the terrain [25].

The major increase of their wealth has motivated Gulf State nations like Qatar and the United Arab Emirates (UAE) to invest in humanitarian activities, including the deployment and sponsorship of search and rescue robotics activities. The UAE Search and Rescue team was one of the first official state-run rescue teams in the world to be equipped with unmanned aerial systems. These are used domestically by response forces, but have also been used by the deployed UAE SAR team during the relief operations after the 2015 earthquake in Nepal to assess the condition of damaged buildings [26]. Next to this operational deployment of rescue robot tools, the UAE has also been sponsoring research in the field through the organizations of challenges

and competitions. In 2015, the UAE organized the first "Drones For Good" international competition [27], which encourages positive applications of drone technology. The first edition of this annual competition was won by a Swiss search-and-rescue drone [28]. Acting as a follow-up of the DARPA challenge, the UAE has launched the Mohamed Bin Zayed International Robotics Challenge (MBZIRC) [29]. This is an international robotics competition, to be held every 2 years with total prize and team sponsorship of USD 5 Million. The first edition is scheduled to take place in 2017. Like in the DARPA challenge, teams will have to complete different tasks, but unlike the DARPA challenge, these tasks are more geared toward collaboration between aerial and ground robots, which will likely steer the developed solutions away from humanoid systems as those used during the DARPA challenge.

2.5. Europe

From an operational side, the European Union Civil Protection Mechanism (EUCPM) is since 2001 fostering cooperation and innovation among national civil protection authorities across Europe. The EUCPM currently includes all 28 European Union member states in addition to Iceland, Montenegro, Norway, Serbia, the former Yugoslav Republic of Macedonia and Turkey. Following the modalities of the EUCPM, member states can request and offer disaster response capabilities (e.g., water pumping capacity for flood relief). Motivated by driving the innovation in disaster management, the European Union Directorate-General for European Civil Protection and Humanitarian Aid Operations (DG-ECHO) is now leading an effort to include the use of robotic tools, focused specifically on unmanned aerial systems, in the EUCPM framework. To this extent, an outdoor demonstration showcasing the benefits of unmanned systems for disaster relief operations was organized in the framework of the 2015 EU Civil Protection Forum [30, 31]. In the wake of this event, DG-ECHO organized a workshop for experts from participating states of the Union Civil Protection Mechanism (UCPM) to discuss the main challenges for the use of unmanned aerial systems in disaster management, in particular their deployment in the context of the EUCPM [32]. The workshop tackled the regulatory, operational, and strategic dimension of the use of unmanned aerial systems for disaster management.

European crisis management agencies have also taken it up to themselves to explore the use of robotic assets, specifically unmanned aerial systems, for managing response operations. They were supported in these efforts by the European Emergency Number Association, which set up a special working group on the topic of "drones," producing an operations manual [33] for emergency services, providing crisis responders a road book on how to best put unmanned aerial systems into operational service.

The operational efforts of the European Union to introduce rescue robots in the field are supported by decades of EU-sponsored research in this domain to develop robotic solutions which can make a difference on the field. One of the larger EU projects on this topic is the ICARUS project, which is the main subject of this book and which is briefly introduced in the next section. First, the following paragraphs discuss some other EU projects which have advanced the scientific research level in the use of robotic tools in each of the different levels (preparedness, response, and recovery) of the disaster management life cycle:

- The ViewFinder project (2006–2009) [34] focused on the assessment phase, developing ground robotic agents operating in chemically contaminated disaster areas to establish whether the ground can be entered safely by human beings.

- The NIFTI project (2010–2013) [35] concentrated on developing methodologies to let humans and ground robots collaborate better, by developing novel human-robot interaction modalities for urban search and rescue robots. A noteworthy achievement of the NIFTI team was a real-life human-robot team deployment in an earthquake area after the 2012 earthquake in Emilia-Romagna region in Northern Italy. Multiple ground and aerial robotic tools were used in order to assess the damage done to several church buildings.

- The AIRBEAM project (2012–2015) [36] developed a situational awareness toolbox for the management of data coming from unmanned aerial systems and space-based assets in the cases of disasters.

- The DARIUS project (2012–2015) [37] focused on reaching effective levels of interoperability such that unmanned systems can be shared between several organizations, by developing a generic ground station with associated standards.

- The TIRAMISU project (2012–2015) [38] considers the use of robotics assets (both ground and aerial robots) for specific types of crisis management operations, namely those where land mines and unexploded ammunitions pose a problem.

- The BerisUAS project (2014–2015) [39] investigated the potential of unmanned aerial systems for marine disaster response operations.

- The R^3 project (2014–2015) [40] aimed to develop a deployment model of robots in disaster management. Besides technical questions such as proper use cases, tactical, operational, and legal issues were also tackled.

- Inspired by the DARPA Challenge, the euRathlon project (2013–2015) [41] organized a competition for rescue robots, requiring a team of land, underwater, and flying robots to work together to survey a disaster scene, collect environmental data, and identify critical hazards. After the final euRathlon event in 2015 (discussed further in chapter 6 of this book), euRathlon transitioned into the European Robotics League for Emergency Robots [42].

- The CADDY project (2014–2016) [43] developed autonomous underwater and surface robots that act as companion to marine search and rescue divers. Note that this is one of the few European projects focusing specifically on marine search and rescue robots, whereas most others target mostly the land and aerial domains.

- The WALK-MAN project (2013–2017) [44] aims to develop a humanoid robot that can operate in buildings that were damaged following natural and man-made disasters.

- The TRADR project (2013–2017) [45] builds on the experience of the NIFTI project for human-robot collaboration in an urban search and rescue context, by building persistent environment models to improve team members' understanding of how to work in the disaster area. TRADR robots were successfully deployed in order to deal with the damage assessment operations after the 2016 earthquake in Amatrice, Italy.

- The RECONASS project (2013–2017) [46] developed a monitoring system, including unmanned aerial systems, that provides a near real time, reliable, and continuously updated assessment of the structural condition of the monitored facilities after a disaster

- The SHERPA project (2013–2017) [47] develops a mix of ground and aerial robotic platform which act as supportive agents to help in alpine search and rescue operations (winter and summer mountain rescue). Key research areas are robustness, autonomy, cognitive capabilities, collaboration strategies, and natural and implicit interaction between the human and the robots.

- The INACHUS project (2015–2018) [48] aims at providing wide-area situation awareness solutions, including novel snake-like robotic agents, for the improved detection and localization of victims trapped inside semi-demolished buildings.

- The Centauro project (2015–2018) [49] aims at the development of a human-robot symbiotic system where a human operator is tele-present with its whole body in a Centaur-like robot, which is capable of robust locomotion and dexterous manipulation in the rough terrain and austere conditions characteristic of disasters.

3. How does the European ICARUS project fit into the development process of search and rescue robots?

As can be noticed in the previous section, there is a vast literature on research efforts toward the development of unmanned search and rescue (SAR) tools, notably in the context of EU-sponsored projects. This research effort stands in contrast to the practical reality in the field, where unmanned search and rescue tools have great difficulty finding their way to the end-users. Notable bottlenecks in the practical applicability of unmanned search and rescue tools are as follows:

- Slow deployment time of the current generation of unmanned SAR tools

- Limited autonomy and self-sustainability of the current generation of unmanned SAR tools, both from a point of view of the robot intelligence and from an energy and mobility perspective

- Limited collaboration between unmanned SAR devices

- Insufficient integration of the current generation of unmanned SAR tools in the C4I equipment used by fire and rescue services

- Insufficient support and training are available for the end-users to learn to use the unmanned tools

- Problems of interoperability of (unmanned SAR) equipment when multi-national crisis management teams need to collaborate on an incident site

The ICARUS project [50, 51] addressed these issues, bridging the gap s the research community and end-users. The ICARUS project was a completely end-user-driven project, where search and rescue workers expressed their operational needs, assisted with the development of solutions and defined and evaluated the developed components. The ICARUS project did not only

focus on the development of tools and services, but also on the integration of these novel tools into the standard operating procedures of the end-users. Indeed, in many cases these integration issues, procedural incompatibilities or absence of legal framework are the main bottlenecks impeding a successful deployment in practical operations and not pure technological issues. ICARUS therefore concentrated also on placing novel technological tools into the hands of the end-users, thereby driving the acceptance and practical use of these tools. These end-user-related aspects of the project are discussed more in detail in the second chapter of this book.

Based on the operational needs of the end-users, the ICARUS project developed robots which have the primary task of gathering data. The unmanned SAR devices are foreseen to be the first explorers of the area, as well as *in situ* supporters to act as safeguards to human personnel. As every crisis is different, it is impossible to provide one solution which fits all needs. Therefore, the ICARUS project concentrated on developing components or building blocks that can be directly used by the crisis managers when arriving on the field. By the end of the project, ICARUS had adapted three aerial robotic systems, two ground robots, and three types of marine vehicles.

On the aerial side, there is a solar aircraft, which beat the world record for continuous flight, staying in the air for a full 81 hours. The plane is 6 meters long, but only weighs 6 kg and fits into a small box when unmounted. It also has another important plus: it can fly at a low altitude, which makes it easier to obtain the necessary flight permits. The second unmanned aerial vehicle is an octocopter, i.e., an aircraft with eight rotors. Equipped with visual and infrared cameras, it can not only produce very accurate 3D maps of the environment for incident mapping but can also drop rescue kits. The smaller third platform is much more autonomous when it comes to taking decisions and navigating as it is designed to enter semi-destroyed buildings where the human controller is likely to lose communication with the device once on the inside. With a very powerful, yet light and power-saving stereo camera sensor on board, it can do 3D reconstruction in real time, a feature crucial for effective indoor navigation. The ICARUS aerial robotics developments are further discussed in the third chapter of this book.

In terms of ground vehicles, the project developed two kinds of platforms. The project's larger vehicle can break a building's wall to clear a passage to the people inside, clear away debris, or position pneumatic poles to stabilize unsound structures. A smaller vehicle that can go inside buildings is equipped with an arm for sensing and grabbing objects, as well as searching for victims. These vehicles are further explained in the fourth chapter of this book.

Finally, the consortium built three platforms for SAR operations at sea: a slower vessel for detection as well as for dealing with incidents close to the harbor, a very fast vessel, and "unmanned capsules," a smaller kind of boat carrying life rafts. The capsules can be deployed from the larger vessels. The faster vehicles get close to the victims, but remain at a safe distance from where the unmanned capsule is deployed, which can propel itself very close to the victim. There, it deploys the self-inflating life raft for the victims to climb on board. The development of the marine robots is explained in the fifth chapter of this book.

In order not to increase the cognitive load of the human crisis managers, the unmanned SAR devices were designed to navigate individually or cooperatively and to follow high-level

instructions from the base station. Seamless interoperability between these different unmanned assets was a key focus point of the project, as further discussed in the sixth chapter of this book.

The ICARUS robots connect wirelessly to the base station and to each other, using a wireless self-organizing cognitive network of mobile communication nodes which adapts intelligently to the terrain and to the available spectrum topology, as detailed in the seventh chapter of this book.

The unmanned SAR devices are equipped with sensors that detect the presence of humans and with a wide array of other types of sensors. At the base station, all the data were processed and combined with geographical information, thus enhancing the situational awareness of the personnel leading the operation with *in situ* processed data that can improve decision-making. All this information is seamlessly integrated in existing information systems, used by the forces involved in the operations, as explained in the eight chapter of this book.

In the world of search and rescue, training is the key. Crisis managers will not use any tool on the field if they have not been extensively trained to use the tool. Therefore, ICARUS concentrated as well on the development of novel training tools, using virtual reality and e-learning in order to provide a quantifiable assessment of the capabilities of the rescue workers to work with the ICARUS robots, as explained in chapter nine of this book.

In order to validate the different ICARUS tools, two main demonstration scenarios were scripted by end-users: an earthquake response scenario and a shipwreck incident scenario. In this manner, an integrated proof-of-concept solution was proposed, evaluated by a board of expert end-users, ensuring that the real operational needs were addressed. Chapter 10 of this book reports on the outcome of these validation scenarios, as well as a real-life deployment of ICARUS tools during a flood relief operation in Bosnia-Herzegovina.

4. Conclusions

As proven by past successes and impressive research efforts around the world, unmanned robotic tools have a great promise to increase the effectiveness of search and rescue operations. However, there are still a large number of bottlenecks which prevent the successful introduction of these unmanned tools on the practical terrain. The European Union ICARUS project has tried to tackle some of these issues by following an approach of tight inter-relation with the end-users and of developing multi-tiered systems, i.e., making systems which are modular up to a certain degree, such that they can do multiple tasks, but not trying to do everything with one system, which would lead to an overflow of requirements. The following chapters in this book describe how this design approach was brought into practice and onto the terrain, even during disasters.

Acknowledgements

Author details

Geert De Cubber[1]*, Daniela Doroftei[1], Konrad Rudin[2], Karsten Berns[3], Anibal Matos[4], Daniel Serrano[5], Jose Sanchez[6], Shashank Govindaraj[7], Janusz Bedkowski[8], Rui Roda[9], Eduardo Silva[10] and Stephane Ourevitch[11]

*Address all correspondence to: geert.decubber@rma.ac.be

1 Royal Military Academy of Belgium, Brussels, Belgium

2 Eidgenoessische Technische Hochschüle Zürich, Raemistrasse, Zürich, Switzerland

3 Technische Univeristät Kaiserslautern, Gottlieb-Daimler-Straße, Kaiserslautern, Germany

4 INESC TEC – Institute for Systems and Computer Engineering, Technology and Science and FEUP - School of Engineering, University of Porto, Porto, Portugal

5 EURECAT Technology Center, Cerdanyola del Vallès, Barcelona, Spain

6 Integrasys SA, Calle Esquilo, Madrid, Spain

7 Space Applications Services NV/SA, Leuvensesteenweg, Zaventem, Belgium

8 Instytut Maszyn Matematycznych, Krzywickiego, Warszawa, Poland

9 ESRI Portugal, Lisboa, Portugal

10 INESC TEC – Institute for Systems and Computer Engineering, Technology and Science and ISEP - School of Engineering, Polytechnic of Porto, Porto, Portugal

11 Spacetec Partners SPRL, Brussels, Belgium

References

[1] De Cubber G, Balta H, Doroftei D, Baudoin Y. UAS deployment and data processing during the Balkans flooding. In: IEEE International Symposium on Safety, Security, and Rescue Robotics; 27-30 October 2014; Hokkaido, Japan. IEEE; 2014

[2] UN - OCHA. INSARAG - International Search and Rescue Advisory Group: Overview [Internet]. Available from: http://www.unocha.org/what-we-do/coordination-tools/insarag/overview [Accessed: 22-November-2016]

[3] UN OCHA. INSARAG Guidelines [Internet]. 2015. Available from: http://www.insarag.org/methodology/guidelines [Accessed: 22-November-2016]

[4] Wagemans R, De Cubber G. RPAS and their challenges. In: INSARAG Team Leaders Meeting; September 2014; Doha, Qatar. International Search and Rescue Advisory Group (INSARAG) 2014.

[5]   International Maritime Rescue Federation. International Maritime Rescue Federation [Internet]. Available from: http://international-maritime-rescue.org/ [Accessed: 22-November-2016]

[6]   Humanitarian UAV Network. UAViators [Internet]. Available from: http://uaviators.org/ [Accessed: 22-November-2016]

[7]   Center for Robot-Assisted Search and Rescue. Roboticists Without Borders [Internet]. Available from: http://crasar.org/roboticists-without-borders/ [Accessed: 22-November-2016]

[8]   Humanitarian UAV Network. UAViators Crisis Map [Internet]. 2009.Available from: http://map.uaviators.org/uaviators/ [Accessed: 22-November-2016].

[9]   Meier P. Digital Humanitarians. CRC Press Taylor & Francis Group, Boca Raton, Florida, United States; 2015. DOI: 10.1201/b18023-2

[10] Meier P, Soesilo D. Drones for Humanitarian and Environmental Applications. 2016. Available from: http://drones.fsd.ch/en/homepage/

[11] Murphy R. Disaster Robotics. MIT Press; Cambridge, USA. 2014. p. 240

[12] Pellenz J, Jacoff A, Kimura T, Mihankhah E, Sheh R, Suthakorn J. RoboCup Rescue Robot League. Lecture Notes in Computer Science. 2015;**8992**:673-685. DOI: 10.1007/978-3-319-18615-3_55

[13] Kitano H, Minoru A, Yasuo K, Itsuki N, Eiichi O. RoboCup: The Robot World Cup Initiative. 1995

[14] Orlowski C. DARPA Robotics Challenge (DRC) [Internet]. Available from: http://www.darpa.mil/program/darpa-robotics-challenge [Accessed: November 23, 2016]

[15] Ackerman E, Guizzo E. DARPA Robotics Challenge Finals: Rules and Course. IEEE Spectrum, New York, USA; 2015XXX

[16] Ackerman E, Guizzo E. DARPA Robotics Challenge: Amazing Moments, Lessons Learned, and What's Next. IEEE Spectrum, New York, USA. 2015

[17] Jacoff AS, Huang H, Messina ER, Virts AM, Downs AJ. Comprehensive standard test suites for the performance evaluation of mobile robots. In: Proceedings of the 2010 Performance Metrics for Intelligent Systems (PerMIS); September 28-29; Baltimore. 2010

[18] Nagatani K,Kiribayashi S, Okada Y, Otake K, Yoshida K, Tadokoro S, Nishimura T, Yoshida T, Koyanagi E, Fukushima M, Kawatsuma S. Emergency response to the nuclear accident at the Fukushima Daiichi Nuclear Power Plants using mobile rescue robots. Journal of Field Robotics. 2012;**30**(1):44-53. DOI: 10.1002/rob.21439

[19] Tadokoro S. Social Implementation of Disaster Robots and Systems - In: IEEE Robotics and Automation Magazine, Tohoku University; New York, USA. September 2015;175-176

[20] Impulsing Paradigm Change through Disruptive Technologies Program/Cabinet Office, Government of Japan. Tough Robotics Challenge (TRC) [Internet]. 2014. Available from: http://www.jst.go.jp/impact/en/program/07.html [Accessed: 24-November-2016]

[21] Qi J, Song D, Shang H, Wang N, Hua C, Wu C, Qi X, Han J. Search and rescue Rotary-Wing UAV and its application to the Lushan Ms 7.0 Earthquake. Journal of Field Robotics. 2015;**33**(3):290-321. DOI: 10.1002/rob.21615

[22] The Korea Herald. Korea, US to Develop Disaster-response Robot Tech [Internet]. 2016. Available from: http://www.koreaherald.com/view.php?ud=20161018000204 [Accessed: 24-November-2016]

[23] Sharkov D. Russian Navy to Develop Arctic Rescue Robots. Newsweek. 2015 [Internet]. Available from: http://europe.newsweek.com/russian-navy-develop-arctic-rescue-robots-318264?rm=eu

[24] Sputnik News. Russian Army Rescue Teams to Get Unique 'Caterpillar' Robot [Internet]. 2016. Available from: https://sputniknews.com/russia/201610291046872437-russia-army-robot/ [Accessed: 24-November-2016]

[25] Tsarichenko SG, Simanov SE, Sidorov IM. Full-scale functional test of special robotics. In: International Scientific and Technological Conference Extreme Robotics; November 24-25, 2016; Saint-Petersburg, Russia. 2016

[26] The National UAE. UAE Search and Rescue Teams Support Nepal Quake Victims [Internet]. 2015. Available from: http://www.thenational.ae/uae/government/uae-search-and-rescue-teams-support-nepal-quake-victims [Accessed: 24-November-2016]

[27] ICT Fund initiated by the Telecommunications Regulatory Authority of the government of the United Arab Emirates. https://dronesforgood.ae/[Internet]. Available from: https://dronesforgood.ae/ [Accessed: 24-November-2016]

[28] The National UAE. Swiss Search-and-Rescue Drone Wins UAE Competition [Internet]. 2015. Available from: http://www.thenational.ae/uae/science/swiss-search-and-rescue-drone-wins-uae-competition [Accessed: 24-November-2016]

[29] Khalifa University. Mohamed Bin Zayed International Robotics Challenge [Internet]. Available from: http://www.mbzirc.com/ [Accessed: 24-November-2016]

[30] European Commission Directorate General for Humanitarian Aid and Civil Protection. European Civil Protection Forum, Brussels, Belgium; 2015. DG ECHO; 2015. p. 76

[31] SpaceTec Partners. EU Civil Protection Forum 2015 RPAS Demonstration and Conference [Internet]. 2015. Available from: http://client.deribaucourt.com/2015-05-06-epcf/ [Accessed: November 2016]

[32] European Commission Directorate General for Humanitarian Aid and Civil Protection. Remotely Piloted Aircraft Systems (RPAS) workshop for Civil Protection experts - Final Report. 2016

[33] O Brien et al. Remote Piloted Airborne Systems and the Emergency Services. EENA Operations Document, Brussels, Belgium. 2015

[34] Baudoin Y, Doroftei D, De Cubber G, Berrabah SA, Pinzon C, Warlet F, Gancet J, Motard E, Ilzkovitz M, Nalpantidis L, Gasteratos A. VIEW-FINDER: Robotics assistance to fire-

fighting services and Crisis Management. In: IEEE, editor. IEEE International Workshop on Safety, Security & Rescue Robotics (SSRR 2009); 2009; Denver, CO. 2009. pp. 1-6. DOI: 10.1109/SSRR.2009.5424172

[35] Kruijff GM, Kruijff-Korbayová I, Keshavdas S, Larochelle B, Janíček M, Colas F, Liu M, Pomerleau F, Siegwart R, Neerincx MA, Looije R, Smets NJJM, Mioch T, van Diggelen J, Pirri F, Gianni M, Ferri F, Menna M, Worst R, Linder T, Tretyakov V, Surmann H, Svoboda T, Reinštein M, Zimmermann K, Petříček T, Hlaváč V. Designing, developing, and deploying systems to support human–robot teams in disaster response. Advanced Robotics, special issue on Disaster Response Robotics. 2014;**28**(23):1547-1570

[36] Bourdache K. AIRBEAM - AIRBorne information for emergency situation awareness and monitoring. In: European Symposium on Border Surveillance and Search and Rescue; November 2014; Heraklion, Greece. 2014

[37] Broatch S. Deployable SAR integrated chain with unmanned systems (DARIUS). In: European Symposium on Border Surveillance and Search and Rescue; November 2014; Heraklion, Greece. 2014

[38] Yvinec Y, Baudoin Y, De Cubber G, Armada M, Marques L, Desaulniers JM, Bajic M. TIRAMISU: FP7-Project for an integrated toolbox in Humanitarian Demining. In: GICHD Technology Workshop; Geneva, Switzerland. 2012

[39] Weyland-Ammeux V, et al. BEtter Response and Improved Safety through Unmanned Aircraft Systems - Berisuas. INTERREG IV A 2 Mers Seas Zeeën, 2 Seas Magazine, Lille, France; 2014

[40] Steinbauer G, Maurer J, Krajnz H. R^3: Request a rescue robot. In: IEEE International Symposium on Safety, Security, and Rescue Robotics (SSRR), 2014; October 2014; Hokkaido, Japan. IEEE; 2014. DOI: 10.1109/SSRR.2014.7017682

[41] Röhling T. euRathlon - An outdoor robotics challenge for Land, Sea and Air. In: International Conference on Intelligent Autonomous Systems (IAS); July 2014; Padova, Italy. 2014

[42] euRobotics Aisbl. The European Robotics League [Internet]. Available from: https://eu-robotics.net/robotics_league// [Accessed: November 2016]

[43] Mišković N, Pascoal A, Bibuli M, Caccia M, Neasham JA, Birk, A, Egi M, Grammer K, Marroni A, Vasilijević A, Vukić Z. Overview of the FP7 project "CADDY - Cognitive Autonomous Diving Buddy. In: MTS/IEEE OCEANS'15 Conference. IEEE; 2015

[44] N.G. Tsagarakis , D.G. Caldwell , A. Bicchi, F. Negrello, M. Garabini, W. Choi, L. Baccelliere, V.G. Loc, J. Noorden, M. Catalano, M. Ferrati, L. Muratore, A. Margan, L. Natale, E. Mingo, H. Dallali, A. Settimi, A. Rocchi, V. Varricchio, L. Pallottino, C. Pavan, A. Ajoudani, Jinoh Lee, P. Kryczka, D. Kanoulas, "WALK-MAN: A High Performance Humanoid Platform for Realistic Environments", Journal of Field Robotics (JFR) (2016)

[45] Kruijff-Korbayová I, Colas F, Gianni M, Pirri F, de Greeff J, Hindriks K, Neerincx M, Ögren P, Svoboda T, Worst R. TRADR project: Long-term human-robot teaming for robot assisted disaster response. KI - Künstliche Intelligenz, German Journal on Artificial Intelligence. 2015;**29**(2):193-201

[46] Sdongos E, et al. A novel & practical approach to structural health monitoring — The RECONASS vision. In: IEEE Workshop on Environmental Energy and Structural Monitoring Systems (EESMS), 2014; September 2014; Naples, Italy. IEEE; 2014. DOI: 10.1109/EESMS.2014.6923261

[47] Marconi L. SHERPA - Smart collaboration between Humans and ground-aerial Robots for imProving rescuing activities in Alpine environments. In: International Conference on Intelligent Autonomous Systems (IAS); July 2014; Padova, Italy. 2014

[48] Athanasiou G, et al. INACHUS: Integrated wide area situation awareness and survivor localisation in search and rescue operations. In: 5th International Conference on Earth Observation for Global Changes (EOGC 2015) and the 7th International Conference on Geo-information Technologies for Natural Disaster Management (GiT4NDM 2015); December 2015; United Arab Emirates. 2015

[49] Schwarz M, Beul M, Droeschel D, Schüller S, Periyasamy AS, Lenz C, Schreiber M, Behnke S. Supervised autonomy for exploration and mobile manipulation in rough terrain with a Centaur-like robot. Frontiers in Robotics and AI. 2016. DOI: http://dx.doi.org/10.3389/frobt.2016.00057

[50] De Cubber G, Doroftei D, Serrano D, Chintamani K, Sabino R, Ourevitch S. The EU-ICARUS project: Developing assistive robotic tools for search and rescue operations. In: IEEE International Symposium on Safety, Security, and Rescue Robotics (SSRR); October 2013. IEEE; 2013

[51] De Cubber G, Serrano D, Berns K, Chintamani K, Sabino R, Ourevitch S, Doroftei D, Armbrust C, Flamma T, Baudoin Y. Search and rescue robots developed by the european icarus project. In: 7th International Workshop on Robotics for Risky Environments; October 2013; Saint-Petersburg, Russia: IARP; 2013. pp. 173-177

User-Centered Design

Daniela Doroftei, Geert De Cubber,

Rene Wagemans, Anibal Matos, Eduardo Silva,

Victor Lobo, Guerreiro Cardoso, Keshav Chintamani,

Shashank Govindaraj, Jeremi Gancet and

Daniel Serrano

Additional information is available at the end of the chapter

Abstract

The successful introduction and acceptance of novel technological tools are only possible if end users are completely integrated in the design process. However, obtaining such integration of end users is not obvious, as end-user organizations often do not consider research toward new technological aids as their core business and are therefore reluctant to engage in these kinds of activities. This chapter explains how this problem was tackled in the ICARUS project, by carefully identifying and approaching the targeted user communities and by compiling user requirements. Resulting from these user requirements, system requirements and a system architecture for the ICARUS system were deduced. An important aspect of the user-centered design approach is that it is an iterative methodology, based on multiple intermediate operational validations by end users of the developed tools, leading to a final validation according to user-scripted validation scenarios.

Keywords: user requirements engineering, system requirements, system validation, design formalisms

1. Introduction

Following the user-centered design approach [1], the needs, requirements, and limitations of end users of a product, service, or process are given extensive attention at each stage of the design process. Compared to other product design philosophies, a big difference with

the user-centered design approach is that user-centered design tries to optimize the product around how users can, want, or need to use the product, instead of trying to force users to change behavior to adapt to the product.

The user-centered design principles state that designers must analyze and foresee how users are likely to use a product and that they should also test the validity of the initial assumptions with regard to the projected user behavior in real-world operational tests with actual users at each stage of the design process. The user-centered design framework can thus be characterized as a multi-stage problem-solving iterative design process. Testing and operational validation during each of the stages of the design process (requirements identification, proof of concept development, prototype development, final product development) is absolutely necessary, as it is often very difficult for the designers of a product to understand intuitively what a first-time user of their design experiences and what each user's learning curve may look like. In the world of search and rescue (SAR), this requirement for intermediate operational testing is even more important than usual, as the SAR environment is so technology unfriendly due to the harsh operating conditions and the dependence of the human SAR workers on the technological tools at his disposal. This explains why the ICARUS project allocated a lot of effort toward the intermediate operational validation of user needs and expectations via realistic trials and even real-life deployments, as explained in Section 5 of this chapter.

Next to the benefits in terms of product design quality of following the user-centered design approach, there is also another key advantage of this paradigm which is of a more social nature and which is often overlooked: user and societal acceptance. Indeed, the acceptance of unmanned tools (such as those developed within the ICARUS project) both by end users and the general society is paramount for the success of the technology. Keeping the end users closely committed is one of the key drivers to increase the acceptance of unmanned search and rescue tools and has therefore been one of the focus points of the ICARUS project.

2. User identification and requirements gathering

A thorough understanding of the end-user community is a key preliminary factor in order to be able to define a correct set of end-user requirements. In the case of ICARUS, the tools are targeted toward the international search and rescue (SAR) community. Within this community, a distinction needs to be made, depending on the terrain where the SAR operations take place, as there exists a separate urban SAR (USAR) and maritime SAR (MSAR) community.

At an international level, the USAR community is organized by the UN via the INSARAG secretariat. INSARAG is a world-wide network of USAR teams and has developed a standardized set of guidelines and established an external classification (IEC) system for USAR teams [2]. As such, INSARAG provides a single point of entry to access nearly the whole international USAR community.

ICAO and IMO are the international entities that coordinate all MSAR global efforts of their member states. The MSAR system has individual components that must work together to provide the overall service. The global MSAR system operationally relies upon states to establish their national MSAR systems. National MSAR services are then integrated with other states to provide world-wide coverage.

The USAR and MSAR communities are quite separate. Therefore, it was required to set up separate user requirement gathering approaches specifically targeted toward each of these communities. For both communities, an iterative information gathering approach was followed, where multiple draft documents were compiled, reviewed, and validated by an end-user board, consisting of members of both the USAR and MSAR communities. Main information sources for these draft documents were personal interviews with key stake-holders, online questionnaires targeted specifically at both communities, and data collected from previous user requirement documents. The draft user requirements were also vali-dated through presentations and discussions at key events where end users were present.

3. Main user requirement

A complete overview of the user requirements for all ICARUS goes beyond the scope of this chapter. Here, we focus on the requirements that are often disregarded by scientists developing robotic systems [3].

3.1. Fast deployment

Unmanned SAR tools that need to be deployed quickly in remote areas must meet the requirements of air transportability, imposing important constraints on the weight and size of all components. Indeed, goods to be transported over the air must fit inside the cargo bay of standard aircraft used for rescue operations. At the moment of a crisis operation, aircraft cargo space is generally very expensive, as many airplanes are demanded in a short period of time. As such, also the size of the package to be transported must be kept to a minimum. In practice, one can conclude that the whole rescue package should fit on two standard euro-pallets, which limit the dimensions to 120 cm × 160 cm × 95 cm. This package must not only contain the robotic tools themselves, but evidently also all the tools to repair them. Moreover, the package must not contain any dangerous goods to avoid problems and delays with customs. High-power batteries, traditionally used for robotic tools, pose a serious problem here, as these are often considered as dangerous goods. Following the INSARAG deployment guidelines, it must be possible to deliver the goods to be trans-ported at the national airport, within 6 hours after getting notice of deployment. Also, important is the total weight of the package, which must of course be brought to a mini-mum. Realistically, the maximum mass for a package can be estimated at 100 kg. This is the maximum weight for a package, such that two humans can still offload it from a cargo plane. If the mass exceeds this number, then a forklift is necessary, which is often difficult to find in crisis areas.

3.2. Manpower requirements

SAR teams are always faced with a massive overload of work. Therefore, it is no easy compro-mise to "sacrifice" people to operate the robotic tools. End users were asked to indicate how many extra team members they could incorporate in their teams for operating unmanned tools. The main conclusion is that no more than two people should be required to operate all the robotic tools.

3.3. Energy requirements

In disaster areas, one cannot count on the availability of a continuous electrical power supply SAR teams generally need to count on their own power sources. Power generators are mostly used for these purposes, and—more and more—also solar panels. Care must be taken that the unmanned tools do not require more electrical power (e.g., for recharging) than can be given by these power generators. The user survey showed that most teams have access to power generators of up to 2 kVA, so this should be regarded as an upper limit for the electrical power draw.

3.4. Water and dust resistance

Very often, SAR teams are working in dusty and wet conditions. Therefore, also the robotic systems should be dust and water resistant. The end users were asked to indicate the desired level of water resistance for the different unmanned platforms, according to the ingress protection (IP) rating code. As a conclusion of this study, the target IP level for outdoor aerial platforms was set at IP53, whereas ground platforms were rated at IP65 and marine platforms were rated at IP85.

3.5. Daytime and nighttime operation

End users want both ground and aerial systems to be able to operate in total darkness. In the case of USAR operations, this requirement is specifically relevant for all indoor platforms, as—in many cases—USAR operations are paused during the night for security reasons. Some USAR teams report on the other hand that the night would be the ideal time for unmanned interventions, as it is calmer and robotic tools could be less constrained by security problems. For MSAR applications, the possibility of doing operation at night is one of the most relevant features and selling points. The reason is that current (manned) MSAR operations almost always need to be halted overnight due to safety concerns. However, unmanned systems could go on throughout the night and could thereby drastically improve the chances of survival of any victims still in the water.

3.6. Autonomy requirements

The level of autonomy to be incorporated in the unmanned systems is always a point of much discussion and is a delicate exercise. Many end users report that in practical SAR operations, the unmanned assets will for the foreseeable future need to be teleoperated for safety and legal reasons. This requirement is in contradiction to the request for easy and human friendly control interfaces and high-level control modalities, which require the incorporation of some degree of autonomy and intelligent autonomous navigation systems. An important factor in this matter is legal issues. Allowing, e.g., unmanned aircraft in civilian airspace is already a sensitive issue in most countries and allowing autonomous aircraft is even more so. Therefore, care must be taken that—at all time—the unmanned aerial platforms can switch to complete teleoperation and act as remotely piloted aircraft (RPA) in order not to limit their deployability in an international context.

3.7. Sensing requirements

End users were requested to prioritize the desired sensing modalities to be installed on the different unmanned systems. The results show clearly that end users value the visual contact with victims (via video cameras) and that geo-referencing of any victims is also deemed to be of high importance. On a second level, infrared and other human detection sensors were selected. On a third level, end users asked for structural 3D mapping capabilities for increasing their situational awareness and also for the presence of a microphone on ground platforms in order to communicate with trapped victims.

3.8. Communication requirements

In crisis areas, the local communication infrastructure is often damaged and largely dysfunctional. Mail and telephone connections often do not work, and Skype chat is one of the most robust services to keep a conversation. Ad-hoc communication tools are therefore clearly required.

3.9. Command and control requirements

Today, there are relatively few hi-tech tools used in an SAR context. This is mainly due to the fact that the crisis environment is extremely technology unfriendly, and SAR workers are therefore reluctant to introduce new technologies in the field. As the crisis managers are under large amounts of stress to carry out a lot of work in a minimum of time, all technologies they are required to use must be extremely user friendly. This means that simple interfacing technologies should be used, hiding most of the background processing tasks from the user, such that the crisis manager only has to give high-level (task) commands.

4. System requirements

Gathering information from the user requirements and the development teams, system requirements and an architecture definition were obtained. The deployment scheme of the ICARUS architecture can be depicted by the scheme of **Figure 1**.

The ICARUS mission planning and coordination system (MPCS) [4] is a system that is deployed at the crisis coordination center and performs the mission planning and coordination activities. Depending on the plan devised at the MPCS, an ICARUS team can perform mission-level activities commanded directly from the crisis coordination center. In the case of a USAR operation, the INSARAG procedures will be followed and this crisis coordination center will be the on-site operations and command centre (OSOCC), where also the local emergency management authority (LEMA) and crisis data providers will input their data and mission objectives. In the event of a maritime SAR operation, this central coordination system would be the national maritime rescue and command center (MRCC).

In the case that the disaster is spanning a wider area, the crisis coordination center will generally divide the crisis area into sectors and then assign incoming SAR teams to the sectors based on the team capabilities and any specific sector needs. The number of sectors can vary enormously based on the extent of the crisis, which means that the coordination system must be very flexible to cope with these very different situations. For reasons of clarity, **Figure 1** sketches a situation with only two sectors, but the architecture is easily extensible. Following this architecture, each sector receives its own robot command and control station (RC2) [4], which connects via the ICARUS communication framework with the MPCS. This communication link will inevitably have to deal with constraints on the amount of data that can be sent over the wireless communication link. The robot operator uses the robot command and control station to control the multiple ICARUS robotic vehicles via the ICARUS communication framework. Some of the ICARUS vehicles are equipped with a robot-victim Human-Machine Interface (HMI) system, enabling disaster victims to send feedback (voice, video) to the RC2, thereby enabling bi-directional communication. At the command station, the local emergency management authority and the crisis data providers and crisis stakeholders interact with the MPCS to input data, enabling the SAR mission planner to assign tasks and missions to the different ICARUS tools via the MPCS. In the field, SAR field teams and first responders are assisted by the ICARUS robots to search for victims and to rescue them. The SAR workers have mobile devices at their disposal, running mobile applications allowing them to read the robot sensor data via the ICARUS communication network and also to contact the RC2 for requesting a change in tasking for the robots.
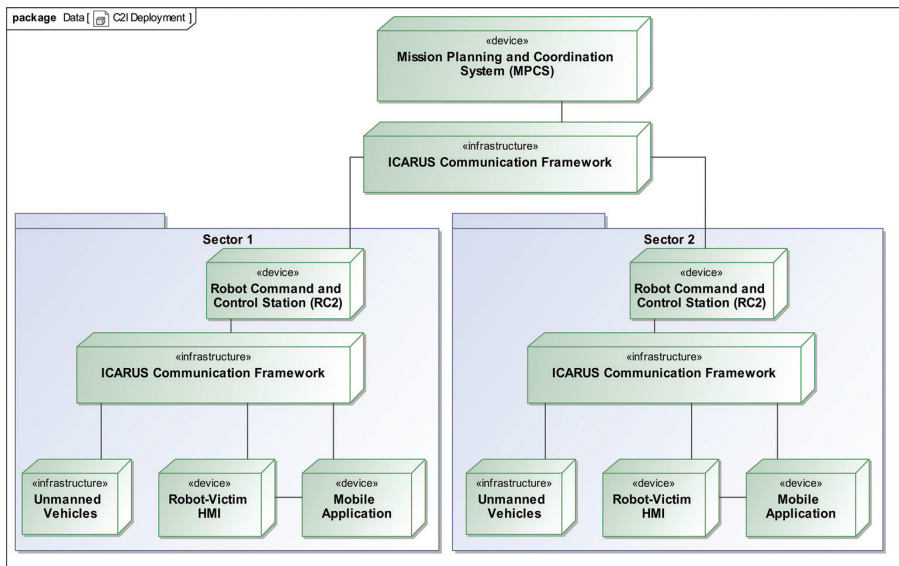

Figure 1. ICARUS deployment architecture (source: ICARUS consortium).

5. Definition of operational validation scenarios

Tools that need to be used by end users in difficult operating conditions need to be validated in a test environment which resembles as much as possible the real-life conditions. It is therefore of the foremost importance that the validation methodology of the robotic search and rescue tools is in line with the real application scenarios as experienced by the end users. To fill this requirement, end users and platform and tool developers together defined a set of use cases for all the tools. A standardized methodology for use-case redaction [5] was followed, which led to a number of use cases. These were then later refined and transformed into validation scenarios. During this process, end users and platform developers were kept in the loop in order to ensure that the proposed scenarios correspond to realistic platform or tool capabilities and to realistic operational conditions.

The approach followed for conceptualizing the validation scenarios was inspired by the approach [6] developed by the National Institute for Standards and Technology (NIST) for developing standardized test methodologies for unmanned ground robots. In this context, each of the validation scenarios consists of three aspects: a detailed scenario, a capability score sheet, and a score sheet for the different metrics (key performance indicators). This makes it possible to qualitatively and quantitatively evaluate the performance of the different tools during the demonstrations.

All scenarios are ordered chronologically, as depicted in **Figure 2** and, when played one after another, form a consistent timeline in line with the demonstration scenarios. Hereby, the leftmost scenario timeline in **Figure 2** corresponds to the USAR demonstration scenario, whereas the rightmost scenario timeline in **Figure 3** corresponds to MSAR demonstration scenario.

Each of these operational validation scenarios will now be briefly introduced [7]:

- The first scenario, C4I_Integration, is a generic application-agnostic scenario where the integration of the higher level ICARUS tools in the existing C4I equipment and procedures of search and rescue workers is validated.

- During the C4I_Mission_Planning scenario, sectors and tasks are assigned to SAR teams by the mission planner. This is done by fusing information from different data sources. The data consists of traditional Geographic Information System (GIS) maps, but also of data from the endurance aircraft which is tasked to map an area.

- During the USAR_Deployment scenario, the USAR teams move toward a sector assigned to them by the mission planner. The main objective of this scenario is to validate the integration of the communication and command and control system and the rapid deployment capabilities. Another goal of this scenario is to validate the developed network management capabilities when confronted with very dynamic team and resource allocations.

- During the USAR_Apartments scenario, the USAR team is assisted by the large unmanned ground vehicle (UGV) and the outdoor unmanned aerial system (UAS). Together, they rescue victims trapped in a semi-demolished apartment building. The main purpose of this scenario is to assess the search and rescue capabilities of the large UGV and the outdoor rotorcraft and their collaborative operation mode.

- During the USAR_School scenario, the USAR team is assisted by the UGV and UAV systems. Together, they rescue victims trapped in a semi-demolished school building. The main purpose of this scenario is to assess the search and rescue capabilities of the small UGV and the indoor rotorcraft and their collaborative operation mode.

- During the USAR_Warehouse scenario, the USAR team is assisted by the UGV and UAV systems. Together, they rescue victims trapped in a semi-demolished warehouse building. The main purpose of this scenario is to assess the search and rescue capabilities of the small and large UGV and the indoor and outdoor rotorcraft and their collaborative operation mode.

- During the MSAR_Air-Air scenario, the MSAR team assesses the situation assisted by the endurance aircraft. Subsequently, they deploy into a sector assigned by the mission planner. The main objective this scenario is to validate the collaborative victim search capabilities of the UAS (both the outdoor rotorcraft and the endurance aircraft) and the integration of the communication and command and control system and the rapid deployment capabilities of the system. Another purpose of this scenario is to validate the command and control and network management capabilities when confronted with dynamic team and resource allocations.

- During the MSAR_Air-Marine scenario, the outdoor rotorcraft searches for victims and autonomously guides an unmanned capsule toward the victim, such that it can deploy and inflate a life raft to save the victim. The main purpose of this scenario is to test the collaborative victim rescue abilities of the outdoor rotorcraft and the unmanned capsules.

- During the MSAR Marine-Marine scenario, the fast unmanned surface vehicle searches for victims and deploys unmanned capsules to save the detected victims. Upon reaching the victims, the unmanned capsules inflate the life rafts. The main purpose of this scenario is to test the collaborative victim rescue abilities of the fast unmanned surface vehicle and the unmanned capsules.

- During the MSAR_Air-Marine-Marine scenario, the outdoor rotorcraft searches for victims and guides the unmanned surface vehicle and the unmanned capsule to the victims. The main purpose of this scenario is to test the collaborative victim rescue abilities of the outdoor rotorcraft, the unmanned surface vehicle and the unmanned capsules.
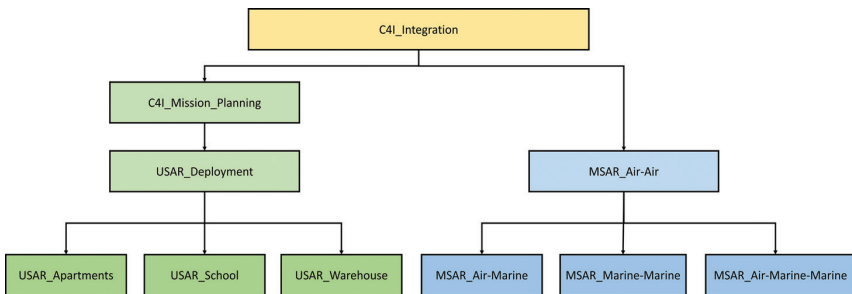

Figure 2. Structure of the different validation scenarios (source: ICARUS consortium).

Figure 3. ICARUS small unmanned ground vehicle demonstrated during the first European unmanned search and rescue end users' conference (source: ICARUS consortium).

Each of the validation scenarios introduced above contains a detailed scenario, which is aligned with the timeline for the demonstrations. Furthermore, each validation scenario contains a list of capabilities that need to be validated, corresponding to system requirements for the different tools. Finally, each validation scenario also includes a score sheet with a number of metrics that are used to quantify the performance of the tools during the operational validation tests. Using this methodology, it is possible to validate the degree to which each of the system requirements has been attained.

In order to clearly indicate the relationship between each and every system requirement and the operational validation scenarios, a traceability matrix was developed, indicating which of the operational validation scenarios apply for each of the system requirements and each of the different ICARUS tools.

As can be noted that the methodology followed here for validation scenario design and quantitative benchmarking has as an objective to strike a balance between on one hand highly standardized (but less realistic) methodologies and on the other hand highly realistic (but less repeatable) methodologies. Following this approach, we provide scenarios and quantifiable validation means that are both scientifically relevant and that ensure the realistic character of the validation trial.

The proposed operational service validation scenarios were incorporated in the intermediate trials and the final demonstration scenarios (which are discussed in Chapter 10 of this book).

The different ICARUS tools were benchmarked and validated during these demonstrations using the scenarios described here. This allowed quantifying the degree of fulfillment for each system requirement set up at the beginning of the project.

6. Operational validation of user needs and expectations

Expressing requirements is often very hard without evaluating the practical operational repercussions of these requirements by doing field tests with the tools to be designed. For this reason, the ICARUS consortium has chosen to organize, in close collaboration with end users, multiple operational field trials already very early stages of the project. At these events, the capabilities of early developments and prototypes were showcased, in order to get valuable feedback from the end users, allowing the end users to re-iterate their requirements and allowing the designers to improve the systems.

In a first phase, initial proof-of-concept prototypes were showcased to potential end users in nonoperational conditions, such that the end users could already have a grasp of the effects and repercussions of the requirements they expressed on the different systems. This early design iteration was performed during the first European unmanned search and rescue end users' conference [8], which was specially organized and dedicated to this subject. **Figure 3** shows an early prototype of the ICARUS small unmanned ground vehicle demonstrated to the audience during this event.

Following the first set of feedback resulting from the demonstration of the prototypes, more and more operational trials were organized, following the scenarios defined in the previous section, where the level of realism was increased, meaning also that the difficulty level for the unmanned platforms was raised.

The operational land trials consisted of exercises of a USAR intervention on one of the training sites used by the Belgian first aid and support team (B-FAST). This site comprises two areas: an area with a rubble field simulating a destroyed structure, with an underground tunnel system and another area simulating a town with skeleton houses useful for indoor training. Evidently, the technical evaluation and improvement of the different developed systems was an important aspect during these intermediate trials, and this will be discussed more in detail in the following chapters for each of the tools separately. However, also very important were the evaluation of non-technical operational considerations, such as fast deployment and safe human robot collaboration. **Figure 4** shows a fast deployment test where the B-FAST team deployed, under command of the team leader, using the full set of ICARUS tools, in order to validate that the use of these tools would not delay the team.

Figures 5 and **6** show a trial where ICARUS aerial SAR tools were collaborating with traditional canine search teams in order to look for survivors on an incident site, in order to assess the advantages and disadvantages of both approaches and their capability of working next to another (which is not a given, as it was previously reported that dogs could be disturbed by

Figure 4. Fast deployment test (source: ICARUS consortium).

Figure 5. Collaboration between aerial rescue robots and human rescue workers (source: ICARUS consortium).

Figure 6. Collaboration between aerial rescue robots and canine rescue assets (source: ICARUS consortium).

the (ultrasound) noises made by aerial robots). This trial turned out to be very successful and showed that dogs and aerial search teams are not only capable of working side by side, but that they are complementary tools, as the dogs were very capable in locating victims hidden on or under the ground in demolished buildings, whereas the aerial tools were very capable of locating victims trapped at a higher level (e.g., on rooftops).

We extensively tested the user friendliness of the developed tools, by training the users to use the ICARUS tools and by putting the ICARUS tools into the hands of the end users during

trials and exercises. **Figure 7** shows a search and rescue worker controlling one of the ICARUS unmanned aerial vehicles with only minimal on-the-spot training, evaluating the user friendliness of the control paradigms and the stability of the platform.

Using robotic tools can lead to new safety hazards, as the use of unmanned aerial systems and heavy ground or marine robots is not without dangers. In an already very dangerous crisis environment, these additional risks must be minimized. To this end, the ICARUS project investigated space management issues and operational interferences between unmanned systems and other actors in the crisis environment. From this analysis, a series of guidelines for safe human-robot collaboration were deduced. These procedures were operationally validated during the different trials, as shown in **Figure 8**.

The sea trials took place near La Spezia, Italy, and Sesimbra, Portugal, with the main goal of validating the solutions developed and of obtaining valuable feedback from the end users. Therefore, the Portuguese Navy assembled an expert panel composed of Navy officers working in areas directly related to MSAR, attending the trials, as shown in **Figure 9**.

Simulated crisis response exercises are extremely useful, but the real operational validation of unmanned technology in search and rescue missions can only be done in a real-life operation. Therefore, the ICARUS team did not hesitate when ICARUS partner B-FAST was deployed to Bosnia in 2014 to help with flood relief operations after massive inundations to send along an expert in robotics and an unmanned aerial system (see also [9] and Chapter 10). The mission was highly successful, providing assistance on the crisis sites not only for several international relief teams [B-FAST, Technisches Hilfswerk (THW), …], but also for the Bosnian Mine Action Centre (BiHMAC). **Figures 10** and **11** show how these relief teams were brought into contact with the new technological tool, provided by ICARUS, in collaboration with the

Figure 7. End user controlling ICARUS unmanned aerial system (source: ICARUS consortium).

Figure 8. Safe collaboration between human search and rescue workers and heavy robots (source: ICARUS consortium).

Figure 9. Expert Navy officers evaluating the performance of the ICARUS marine tools (source: ICARUS consortium).

Figure 10. Collaborator of the Bosnian Mine Action Centre being trained during operation on the use of unmanned aerial systems for mine risk mapping (source: ICARUS consortium).

Figure 11. Operatives of the German relief team "Technisches Hilfswerk" (THW) being trained during operation on the use of unmanned aerial systems for damage assessment (source: ICARUS consortium).

TIRAMISU project [10]. As we were during the mission tightly integrated with these end users and their procedures, this provided a deep insight into their requirements, procedures, indicating also the bottlenecks toward the integration of unmanned systems in the standard operating procedures of the search and rescue workers.

7. Conclusions

Throughout the ICARUS project, end-user engagement was one of the key focus points, as we realize that this was the main driver for acceptance of the technologies developed within the project and therefore also for the successful introduction of these tools on the terrain. A user-centered design was adopted, as discussed in this chapter. This led to user requirements and the definition of user-scripted operational validation scenarios for the ICARUS tools. Following the formalism of iterative user-centered design, multiple intermediate validation trials were organized where the ICARUS systems were validated in more and more realistic environments. A lot of attention was paid not to validate only the pure technical capabilities of the systems, but also the very important nontechnical aspects like human-robot collaboration, safe and legal operation, and rapid deployment.

Acknowledgements

Author details

Daniela Doroftei[1]*, Geert De Cubber[1], Rene Wagemans[2], Anibal Matos[3], Eduardo Silva[4,5], Victor Lobo[6], Guerreiro Cardoso[6], Keshav Chintamani[7], Shashank Govindaraj[7], Jeremi Gancet[7] and Daniel Serrano[8]

*Address all correspondence to: daniela.doroftei@rma.ac.be

1 Department of Mechanical Engineering, Royal Military Academy of Belgium, Av. De La Renaissance, Brussels, Belgium

2 Belgian First Aid and Support Team, Brussels, c/o S1.1 Rue des Petits Carmes, Brussels, Belgium

3 INESC TEC – Instituto de Engenharia de Sistemas e Computadores, Tecnologia e Ciência and Faculdade de Engenharia da Universidade do Porto, Campus da FEUP, Rua Dr. Roberto Frias, Porto, Portugal

4 INESC TEC – Instituto de Engenharia de Sistemas e Computadores, Tecnologia e Ciência Campus da FEUP, Rua Dr. Roberto Frias, Porto, Portugal

5 Instituto de Superior de Engenharia do Porto, Rua Dr. António Bernardino de Almeida, Porto, Portugal

6 Escola Naval, Rua Base Naval de Lisboa, Almada, Portugal

7 Space Applications Services NV/SA, Leuvensesteenweg, Zaventem, Belgium

8 Eurecat Technology Center, Av. Universitat Autònoma, Cerdanyola del Vallès, Barcelona, Spain

References

[1] Greenbaum J, Kyng M, editors. Design at Work – Cooperative design of Computer Systems. Lawrence Erlbaum; Hillsdale, NJ, USA; 1991

[2] UN – OCHA. INSARAG – International Search and Rescue Advisory Group: Overview [Internet]. Available from: www.insarag.org [Accessed: November 22, 2016]

[3] Doroftei D, Matos A, De Cubber G. Designing Search and Rescue Robots towards Realistic User Requirements", Applied Mechanics and Materials, Iasi, Romania; Vol. 658, pp. 612-617, 2014

[4] Govindaraj S, Chintamani K, Gancet J, Letier P, Van Lierde B, Nevatia Y, De Cubber G, Serrano D, Bedkowski J, Armbrust C, Sanchez J, Coelho A, Palomares ME, Orbe I. The ICARUS project – command, control and intelligence (C2I). In: Safety, Security and Rescue Robots; October 2013. Sweden: IEEE; 2013

[5] Cockburn A. Writing Effective Use Cases. Addison-Wesley; Boston, MA, USA; 2000

[6] Jacoff A, Messina E, Huang HM, Virts A, Downs A, Scrapper C, Norcross R, Schwertfeger S, Sheh R. Evaluating Mobile Robots Using Standard Test Methods and Robot Performance Data. Technical Report. National Institute of Standards and Technology; Gaithersburg, MD 20899, United States of America; 2012

[7] Doroftei D, Matos A, Silva E, Lobo V, Wagemans R, De Cubber G. Operational validation of robots for risky environments. In: 8th IARP Workshop on Robotics for Risky Environments; January 2015. Lisbon, Portugal: IARP; 2015

[8] Doroftei D, editor. Proceedings of the First European Unmanned Search and Rescue End-Users' Conference. Brussels, Belgium: SpaceTec Partners; 2014. Available from: http://www.spacetecpartners.eu/icarus-darius-event ed

[9] De Cubber G, Balta H, Doroftei D, Baudoin Y. UAS deployment and data processing during the Balkans flooding. In: IEEE International Symposium on Safety, Security, and Rescue Robotics; 2014

[10] Yvinec Y, Baudoin Y, De Cubber G, Armada M, Marques L, Desaulniers JM, Bajic M. TIRAMISU: FP7-Project for an integrated toolbox in Humanitarian Demining. In: GICHD Technology Workshop; Geneva, Switzerland; 2012

Unmanned Aerial Systems

Rudin Konrad, Daniel Serrano and Pascal Strupler

Additional information is available at the end of the chapter

Abstract

Unmanned aerial platforms are a means to gather efficiently valuable aerial information to support the crisis manager for further tactical planning and deployment. They can provide continuous support to the coordinators and operators by scanning blocked sectors or establish an communication network. This chapter describes how aerial platforms were tailored to search and rescue (SAR) requirements, including the localisation and tracking of victims. In order to meet the end user demands, complementary platforms are proposed. A small long-endurance solar aeroplane is used to provide the largest and fastest area coverage at the highest view, and therefore enabling the mapping functionality and potential detection of victims with operation times span up to a day. Complementary to the aeroplane, two rotary-wing systems were deployed. A large coaxial-quadrotor was used for outdoor delivery task and detailed close range inspection. Its ability to fly close to the terrain enables a thorough search for victims in a well-defined sector. A smaller multicopter was used for inspection of the indoor environment. It is able for victim detection in collapsed buildings. Thus, autonomous functionality for precise localisation and positioning was developed to decrease the operator workload.

Keywords: unmanned aerial systems, drones, search and rescue

1. Introduction

The field of the micro electro-mechanical system (MEMS) technology enables small and light-weight, yet accurate inertial measurement units (IMUs), inertial and magneto-resistive sensors. The latest technology in high-density power storage offers high-density battery cells. Thus, allowing lightweight unmanned aerial systems (UASs) with increased flight time suitable for varying missions. So far, UASs have been used mainly for military applications, for reconnaissance but also in engagement. However, increasing attempts are being made to use such

systems in non-military scenarios. Within the ICARUS project, several UAS platforms are being developed to support human teams in disaster scenarios as part of an integrated set of unmanned search and rescue (SAR) systems. While there exist a wide range of aerial platforms equipped with basic functionality, the UASs need to be adapted with functions specific to the ICARUS framework and its scenarios.

The autonomous UASs for SAR applications are suited as remote sensors for mapping the disaster area, inspection of the local infrastructure, efficient localisation of people, and as tools for fast interaction with them. As an additional capability, the UAS tools will be complemented with a communication relay functionality to establish its own communication network in case the local infrastructure is broken. To fulfil these missions, the UAS systems will have to feature accurate on-board localisation and control while still requiring minimum power. The power requirement is a crucial role for UASs in order to allow for sufficient operation time carrying the on-board payload. The systems need to work autonomous with minimal human interaction, allowing human forces to be focused on other tasks. Furthermore, the project intends to provide solutions for adapting to unknown or even dynamically changing environments: as they operate close to people, it is necessary for the platforms to function robustly without hitting anything or anyone.

All necessary payloads for the SAR missions need to be identified and integrated with the platforms, while still meeting the tight specifications on endurance and manoeuverability. This consists of additional sensors and devices for flight autonomy and integration into the ICAURS framework.

2. UAS applications for SAR missions

SAR missions can benefit from the use of UASs in many distinct ways. Their unique ability to reach higher altitudes from ground can be useful to provide means, other robots or humans on ground can never achieve. Within the ICARUS project, five distinct functionalities of UASs are proven to be useful

1. Mapping/sectorisation: A UAS can provide operators on-site and even higher level coordinators with valuable aerial information over an extended period of time and at a large scale in an efficient period of time. This information can be used for planning and risk assessment. A direct video stream to the operator can give them valuable live information about the situation. Furthermore, the aerial data can be processed to build a 3D map of the scanned area or building.

2. Victim search: Search for victims is one of the key elements of the ICARUS system. Using to a large extent the on-board thermal cameras, a large area can be scanned and potential victim locations are forwarded to the operator automatically to be verified.

3. Target observation: The UAS provides the operator/coordinators with a pair of remote eyes in the sky. A specified location can be watched for an extended period of time without having to care about low-level control of the UAS. For the long-endurance UAS intrinsically incapable of keeping the camera view constant over time, a circling of the target will be the

default procedure. The target to be observed can also involve assessment of the structural integrity of buildings both from the outside or the inside. In addition, the captured and registered images can also be used for mapping and sectorisation.

4. Delivery: Items possibly useful to be delivered by the UAS consist of, for example, bottles of water, medical packages and potentially inflatable buoyancy aids. The UAS must be capable of carrying and deploying those items to a user-specified location.

5. Communication relay: In a distributed network of robotic and human agent, communication plays a key role. UASs naturally offer good visibility to a large area and are thus suited for acting as communication relay when integrated in a common network. This functionality differentiates itself from the previous ones, since the request for acting as a communication relay will typically be received when UAS platforms are on other missions (but it is of course not limited to this situation). In this case, the prioritisation of the communication relaying (i.e. possible interruption/relocation) over the current mission is handled by the coordinator or the UAS operator. A flight plan may be suggested automatically providing coverage necessary for performing the relaying task.

In order to provide efficient tools for the defined functionalities, a set of heterogeneous UASs have been developed. The goal of all UAS platforms is to efficiently gather information about the disaster area and possible victims, to provide initial life support and interaction with victims, assist the SAR teams and ground robots with additional information, and acting as communication relay in case the local infrastructure is broken. During a SAR mission, different level of precision of the provided information is needed, which leads to the developed platforms with different types of area coverage and speeds.

The first platform is a long-endurance fixed-wing UASs, as shown in the top image of **Figure 1**. This UAS flies at low altitudes in the airspace of 150–200 m above the ground. It can provide aerial images both from visual and thermal cameras. With its long endurance ranging from several hours up to a few days and its autonomous capabilities, the long-endurance UAS is able to cover large areas in a short period of time. Thus, it is most useful in the planning phase of the SAR teams. It can be started at the base of operation (BOO) before reaching the disaster side with the SAR team. Its main purpose is to complement the aerial images gathered from satellites for a first assessment of the disaster side including pointing at potential positions of victims. It has a fast deployment time of a few hours and due to its hand launch capability and autonomy it is easy to operate. The (potential fleet of) long-endurance UAS can be given a scan area to cover by aerial images. It will autonomously scan the area and return to the BOO to deliver the image data for further processing. This information can be used as a first overview of the level of destruction at the disaster side and as an assessment of the local infrastructure, for example, the roads leading to the disaster side for moving to and installing the forward base of operation (FBOO). Furthermore, this UAS is an ideal candidate to act as a communication relay with its high ground clearance, connecting the different SAR teams at a further stage will still providing aerial information.

The second platform is a large outdoor quadrotor shown in the bottom left image of **Figure 1**. This UAS is used for detailed observation and for gathering thermal and visual aerial imageries at low altitudes up to 100 m above the ground. It is packed within a compact box and can

Figure 1. UAS fleet within ICARUS. Top: AtlantikSolar from ETH. Bottom left: AROT from EURECAT. Bottom right: Indoor multirotor from Skybotix (source: ICARUS consortium).

be transported to and used at the FBOO or be used during displacement of the SAR team. It is able to operate autonomously while mapping in detail a given area or it can be directly piloted over the remote command and control system (RC2) if the operator wants to assess specific details. It is able to map a short area of interest with a much higher resolution than the fixed-wing UAS. Furthermore, due to the lower altitude it is able to confirm the possible locations of victims found by the fixed-wing UAS and search for and detect additional victims in that area. Since the quadrotor is able to hover it is suited to map walls of buildings for structural integrity checks. Furthermore, it permits to interact with victims to assess their health and with the help of the integrated delivery system to provide water or first aid kits to them.

The last platform is the indoor multirotor, as shown in the bottom right of **Figure 1**. This multirotor with a small footprint can be used to inspect buildings from the inside whose structural integrity might be compromised. In comparison to the unmanned ground vehicles (UGVs), the aerial multirotor has the advantage of easily accessing the buildings without being blocked by possible debris inside, as long as there is aerial clearance. Furthermore, it can easily climb different floors for inspection. This UAS assists the SAR teams to assess the structural properties and mapping of the building without risking the live of a human rescue team member. As there is in general no map of the inside available and there might be some debris blocking the paths, the UAS must be piloted by an operator over the RC2, while a video stream is fed back to enable a first person view (FPV) flight. Since the operator will not have a complete oversight of the path, the UAS needs to be aware of potential obstacles and avoid them if necessary.

All those platforms have a very distinctive purpose and complement each other to from a complete set of aerial robots for different possible SAR scenarios. They operate at different locations and altitudes for airspace separation and collision avoidance between the UAS.

3. UAS mechanical design

As discussed, the UAS fleet with its sensors can provide a great deal of situational awareness for the SAR teams. There exists a great variety of available drones in the market. However, they still have just limited autonomy due to the limited sensor capabilities and thus need a constant supervision of a trained pilot. The main task within the ICARUS project are thus to extend the autonomy and situational awareness of the systems to help the SAR teams within their missions with limited human interaction.

3.1. Visual sensor payload

Since all the platforms have similar requirements for the visual sensor payload, a common visual sensor payload was chosen. This reduces overall development and maintenance. This visual inertial (VI) sensor (see **Figure 2**) combines visual information from up to four visual and/or thermal cameras with the information from the inertial measurement unit (IMU). This forms a complete measurement set for vision-based mapping and localisation. Since the resolution and weight constraints are not yet met by the thermal camera development in WP 210, an FLIR Tau 2 thermal camera is used over all platforms. The sensors are hardware synchronised for tight fusion in the image processing algorithms. It is suitable for sensing and performing on-board processing for simultaneous localisation and mapping (SLAM) as well as application-oriented algorithms (cartography, victim detection). The core consists of a field programmable gait array (FPGA) for visual pre-processing as well as an processing unit (Kontron COM express module) that can be exchanged and allows for the use of standard tools (e.g. install a standard Linux operating system and run the vision algorithms). The unit was designed for low-power and low-weight constraints such as use on-board small UAS. The mass amounts to 150 g, while the power consumption is of around 10–15 W.

The sensor unit can be used for accurate pose estimation and mapping at real-time in all six dimensions (position and orientation). The integrated FPGA can provide raw visual data as well as pre-processed visual data such as visual keypoints used for SLAM and mapping. The
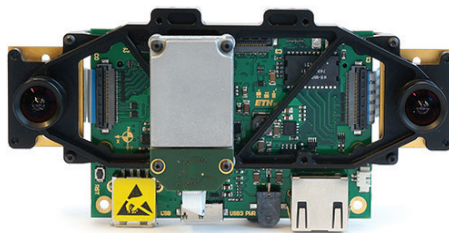

Figure 2. VI sensor hardware with two visual cameras and mounted IMU (source: ICARUS consortium).

team of the swiss federal institute of technology in Zurich (ETHZ) has verified the mapping capabilities of the VI sensor in GPS-denied environments and thorough analysis may be found in Ref. [1].

The raw and processed information from the VI sensor can be used for the mapping, victim search, as well as for the target observation algorithms. Furthermore, increased control performance can be achieved using the improved pose estimation.

3.2. AtlantikSolar

AtlantikSolar from ICARUS partner ETHZ was chosen as the fixed-wing long-endurance UAS. It is a small and lightweight solar-powered UAS. Concerning solar UAS, prototypes of different sizes and wingspans have been successfully operated by NASA, Quinetiq, Airbus, and others. The integrated solar technology increases the overall flight time to even perpetual flights. This makes such UAS a perfect candidate for extended information gathering of large-scale areas, and acting as an airborne communication relay.

The difference between the AtlantikSolar UAS (as shown in **Figure 3**) compared to others is its capability to fly at low altitudes and its transportation and fast deployment capabilities due to its small size and weight. The UAS has a wingspan of 5.6 m and an overall weight of 7.5 kg. Due to the small weight, it can be launched by hand and thus can be deployed at almost all wide open locations without the need of an intact airstrip. The main wing consists of three parts, which can be disassembled for transportation. The ribs of the main wing are built out of balsa wood. These profile giving structures are interconnected by a spar made out carbon-fibre tubes, which runs from wingtip to wingtip. This results in a lightweight structure and allows the transportation of the necessary battery packs with a total weight of 3.5 kg and with capacity of 34.5 Ah at a nominal voltage of 21.6 V. The batteries are stored within the carbon-fibre tubes throughout the main wing. The whole top surface of the main wing is used to embed solar modules. Flying at a nominal speed of 9.5 m/s it is capable of flying up to 10 days at appropriate weather conditions. The nominal power usage is around 60 W while generating up to 250 W at noon with the solar panels. The complete specifications can be found in Ref. [2]. The long-endurance capability was shown with a successful flight demonstration to break a new world record of over 81 hours continuous flight by travelling 2316 km in July 2015 in Switzerland.

The UAS autopilot is based on the open-source Pixhawk project [3], which was adapted for autonomous waypoint-based navigation. It is equipped with a sensor suite consisting of an inertial measurement unit (IMU), a magnetometer for determining the heading of the UAS, a global position system (GPS) device, a pitot tube measuring the airspeed and a static pressure sensor. Those sensors can be fused together to estimate the pose of the UAS at all times as described in Ref. [4]. An additional sensor payload (SensorPod) has been mounted at the wing and incorporates the VI sensor together with the communication system. The sensor payload consists of an atom motherboard with the Linux operating system running ROS. The ROS interface was used for running the vision algorithm as well as communicating over the adapted joint architecture for unmanned system (JAUS) protocol [5] within the ICAURS mesh.
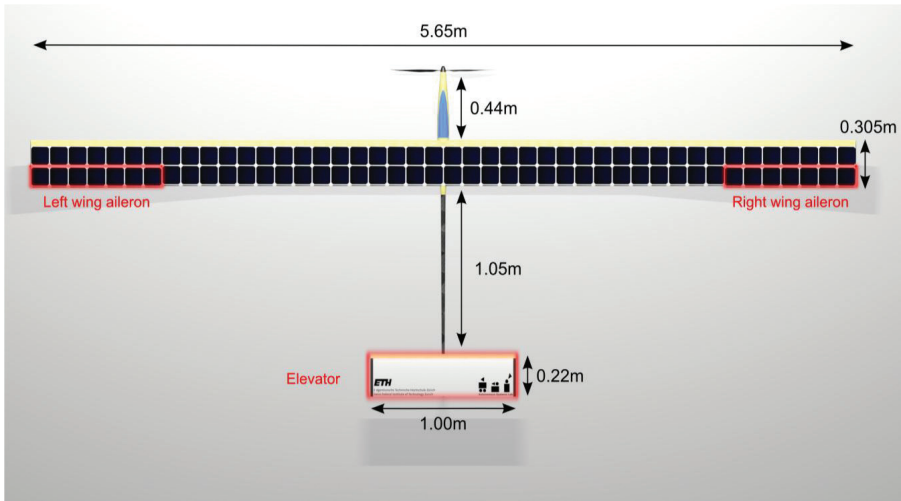
Figure 3. Specification of the AtlantikSolar UAS (source: ICARUS consortium).

As depicted in **Figure 4**, the operator may control the UAS either directly by radio control or by the ground control station (GCS). The manual control is intended for safety reasons only, in normal operation, the UAS can be controlled over the GCS by providing a flight path. Two different long-range communication channels are integrated for the data link to ensure connectivity throughout the whole operation. A high-bandwidth link using the 5 GHz Wi-Fi technology for the camera transmissions, as well as a long-range low-bandwidth communication device for control and operation of the UAS.

Towards complete autonomy, the AtlantikSolar UAS implements path-planning algorithms that address the problem of area coverage and exploration. The algorithm is tackling the problem of covering a large area, taking into account the dynamics of a fixed-wing UAS together with a fixly mounted camera. The implemented algorithm is based on sampling-based methods and is able to search for the optimal path that ensures full exploration of a given area in minimum time. The implemented framework respects the non-holonomic constraints of the vehicle. Results and detailed explanations can be found in Ref. [6]. The framework has been successfully tested in multiple cases including the ICARUS public field trials in Barcelona/CTC and Marche-en-Famenne in Belgium. The results can be seen in **Figure 5**, where the generated map of the Marche-en-Famenne trial site is shown together with the optimised path for total area coverage. Thus, the operator only needs to provide a desired area to be covered, while the UAS is planning fully autonomous the whole mission and returns for delivering the gathered information.

Finally, the UAS is able to detect possible locations of the human victims using both the thermal and visual camera information provided by the VI sensor (see **Figure 6**). Since the human body temperature is generally higher than the surrounding, the thermal imagery can be used to detect humans.
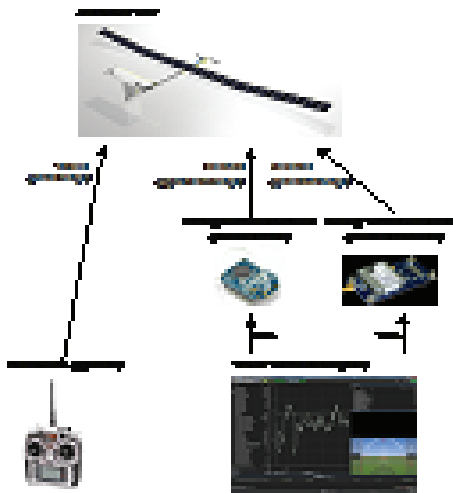
Figure 4. Communication structure of the AtlantikSolar UAS (source: ICARUS consortium).

Figure 5. Optimal path planning for the AtlantikSolar UAS. The path is optimal to cover the trial site in Marche-en-Famenne. The images are used to form the shown map (source: ICARUS consortium).
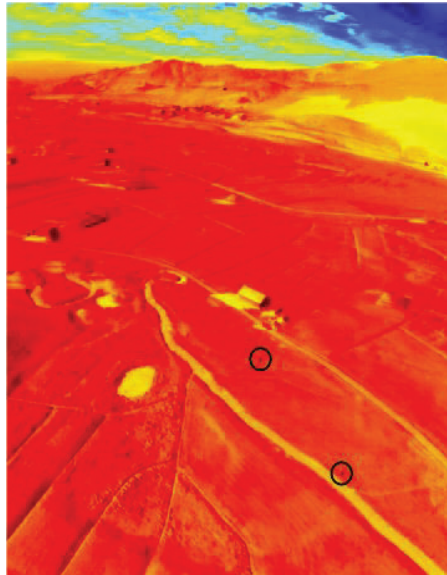
Figure 6. Victim detection with the UAS. Possible locations of human bodies are marked in the picture and send to the RC2 (source: ICARUS consortium).

Due to the restricted resolution of the thermal camera, the visual camera can be used to reference to found location. The algorithms work by subtracting the background from the thermal image and comparing the resulting regions of interest with known human features. A complete explanation of the algorithm can be found in Ref. [7]. The victims are tracked over time by the UAS, as long as they stay within the camera coverage, and their location is sent back and displayed at the RC2.

3.3. LIFT quadrotor

The lightweight and integrated platform for flight technologies (LIFT) coaxial-quadrotor developed by EURECAT is a hovering UAS with vertical take-off and landing capabilities (VTOL), capable of carrying out different tasks in the context of SAR operations. Quadrotor like the LIFT is a popular configuration along the VTOL UAS. Its design is suitable for robust hovering and omnidirectionality. Further, it can generate high torques on its main body for highly agile manoeuvres. The hovering capability and its ability to lift a certain payload of the Eurecat quadrotor complements the functionality of the AtlantikSolar UAS. It is able to fly closer to the ground for detailed mapping, lock and approach on potential victims as well as delivering goods to them. It can be controlled by the operator to give a steady aerial images feed due to his hover capability.

The LIFT (see **Figure 7**) is a large coaxial-quadrotor with a weight of 4.3 kg. The length from shaft-to-shaft is 0.86 m and on each end are two propellers in a coaxial configuration. The

propellers can lift a maximum weight of 9.6 kg. The main frame consists of carbon fibre tubes connected at the centre with the autopilot unit at top and a variable payload attachment at the bottom between the landing gear. The operational altitude is between 25 and 100 m above the ground. Two lithium polymer batteries with a capacity of 10 Ah at a nominal voltage of 21.6 V power the UAS. The flight time depends on the used payload and can be up to 30 minutes before replacing the batteries. It can be safely stored in a box to transport the UAS to the destination point.

The UAS autopilot architecture is similar to that of the AtlantikSolar UAS. A pixhawk autopilot is combined with a netbook processor running Linux and the ROS operating systems for communication in the ICARUS mesh and for running the vision algorithms. The UAS is equipped with state of the art proprioceptive (accelerometers, gyroscopes, and altimeter) sensors used to stabilise the attitude of the UAS within the autopilot explained in Ref. [8]. Using the GPS measurements, the position of the UAS can be controlled. Additionally, the UAS is equipped with perceptive (VI sensor and range) sensors for enhanced state estimation. The platform can fly autonomously or to be piloted remotely. It has an advanced set of sensors to carry out different tasks in the context of SAR operations, such as first aid kit delivery, victims search outdoors, terrain mapping and reconnaissance. It has several communication links, starting from a radio control link to directly control the UAS. This link has an override mechanism to take over the UAS in the case of emergency. The primary communication link is used to send basic information between the GCS and the UAS such that that it can be monitored constantly. Further, the ICARUS communication link is incorporated inside the payload to connect to the ICAURS mesh using the Wi-Fi connection. It can be operated manually up to fully autonomously following a desired path within the GCS or the RC2.

Using the VI sensor with the thermal camera as payload, LIFT is able to map small areas and buildings from the outside. Compared to the fixed-wing UAS it can fly close to the buildings to gather detailed visual information. This information can be used to assess the structural integrity of the buildings. The operator can precisely guide the UAS to get to and lock on specific locations or targets for further investigation using the live video stream. The mapping capability was shown in the Marche-en-Famenne trial site where two buildings are entirely mapped, as shown in **Figure 8**.

Figure 7. The Eurecat coaxial-quadrotor with the camera payload (source: ICARUS consortium).

Figure 8. Map of two buildings from the Marche-en-Famenne trial site created by the Eurecat quadrotor (source: ICARUS consortium).

The visual and thermal camera can be used as well for victim detection. Detected victim positions can be sent to and displayed at the RC2. The aerial capability of the UAS is beneficial to detect humans within a forest or rubble field where humans or robots on ground can only move slowly. Compared to the fixed-wing UAS, LIFT has the advantage to fly at lower altitudes, to hover at a certain point, and to move omnidirectional. Thus, it can cover an area in much more detail and from different specific views to increase the probability to find the victims. One of such potential scenarios is shown in **Figure 9**, where the UAS is capable of detecting a human in the forest partially covered by a tree. This also shows the advantage of using the thermal
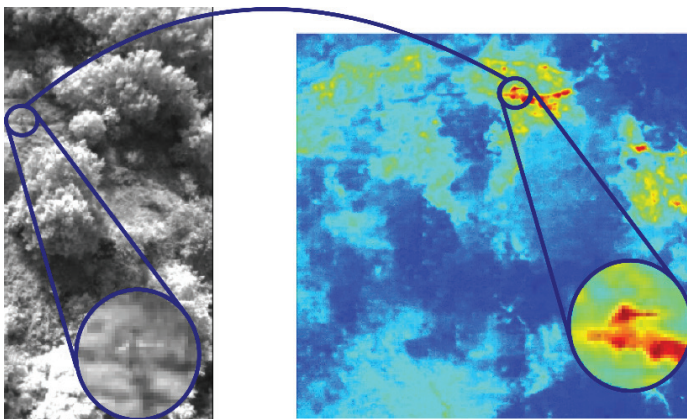
Figure 9. Victim detection using the Eurecat quadrotor. The victim lying in the forest and partially covered by a tree is found using the thermal images (source: ICARUS consortium).

compared to the visual camera, where the victim is barely visible. The people detector takes advantage of histograms of oriented gradients (HOGs) using a sliding window approach in the images. One of the main reasons to use the HOG human detector is that it uses a 'global' feature to describe a person rather than a collection of 'local' features, that is, that the entire person is represented only by a single feature vector. Another important feature is that this classifier is already trained. However, the training data set is limited to some people postures and camera orientations. One drawback of the HOG detector is that people need to have a relative big size in the image (around 64 × 128 pixels), failing when a person occupies a small part of the image (a common situation for aerial images). For this reason, a region-growing algorithm based on temperature blobs is implemented to search for victims.

Finally, the UAS payload can be exchanged for a delivery mechanism. The delivery mechanism is based on an electromagnet as a release mechanism to deliver the goods. Attaching a magnet to the delivery kit, it can be easily connected to the UAS. As soon as the UAS reaches the destination point, the operator can disable the electromagnet and deploy the kit, as shown in **Figure 10**. It is used to help injured but responsive victims by providing them with necessary supplies and treatment until the ground rescue teams arrive.

Figure 10. Eurecat quadrotor delivering a first aid kit to an injured victim (source: ICARUS consortium).

3.4. Skybotix multicopter

The skybotix multicopter is a hexacopter with a small footprint. As the LIFT, it is capable of VTOL flights and hovering. However, it is mainly used to inspect the inside of buildings. With its small footprint it is able to enter building by small openings like open windows or doors. The platform is chosen since it can hover robustly not being sensitive to wind and other disturbances, while still being able to fly through narrow passages. Capable of flying indoors, it can navigate and help analysing the structural integrity of the building from the inside and search for possible humans trapped inside the building.

The Skybotix multicopter, shown in **Figure 11**, is a modified version of the AscTec firefly with a weight of 1.4 kg including the additional sensor payload of 420 g. It has six propellers in a hexagonal configuration with a radius of 0.66 m and a height of 0.17 m. The soft propellers are used to not harm potential humans. The UAS has integrated some fault tolerance, since it can even fly with only five propellers. The propellers are connected with carbon fibre tubes to the centre of the main body. A battery is powering the UAS with a capacity of 4.9 Ah at a voltage of 12 V. It has a maximum flight time of 15 minutes before the battery needs to be replaced. The upper part of the main body encapsulates the proprioceptive sensor and microprocessor for control, whereas the below the VI sensor, communications, and an additional CPU are mounted for mapping and integration into the ICARUS mesh.

The UAS can be flown manually over the dedicated remote control link or over the RC2. It has many different modes of operation starting from attitude stabilised mode to fully position assist mode. Since the indoor environment is not known a priory and the environment could be cluttered, an operator has to constantly fly the UAS by providing translational velocity commands in the position assist mode. A constant video stream provides a feedback for the operator. The

Figure 11. Skybotix multicopter flying through a window for inspecting a building from the inside (source: ICARUS consortium).

basic idea of the obstacle avoidance is to constrain the commanded velocity vector of the UAS by overloading a repelling velocity. This is done by using a potential field around the obstacles generating a repelling velocity, which increases in magnitude while decreasing the distance between the UAS and the obstacle. The tracked velocity command is thus the average of the user set point together with all repelling velocities of all obstacles in the vicinity. Although the user commands can be given as velocity set points, the UAS itself is controlled in position to neglect drift in position as long as there is no user input to the system even in the presence of external disturbances such as wind gusts. A detailed explanation of the control can be found in Ref. [9].

In order to perform obstacle avoidance and UAS navigation, the UAS is required to have a notion of its surrounding. To this end, the VI sensor is used to generate depth images, which are subsequently incorporated into a 3D voxelgrid. The corresponding depth image is estimated using a time-synchronised stereo-image pair from the VI sensor. A block-matching algorithm is used to find correspondence objects in both image frames that can be triangulated knowing the baseline between both cameras. The depth images are only estimate once the UAS has moved to far outside the known terrain and a new pose key frame must be generated, to avoid short-term drifts in the localisation and map generation and to reduce the computational burden. Using the pose key frame, the depth image can be transformed into a 3D point-cloud. Since this point-cloud can quickly become untrackable for large areas, the information must be compressed inside an efficient grid-based map. Within this research, the OctoMap mapping framework is used which results in an environmental representation as shown in **Figure 12**. This representation used a resizable grid, where each node represents a probability of being occupied. All the node probabilities are updated by incorporating the new point-cloud with its associated noise model uncertainty.
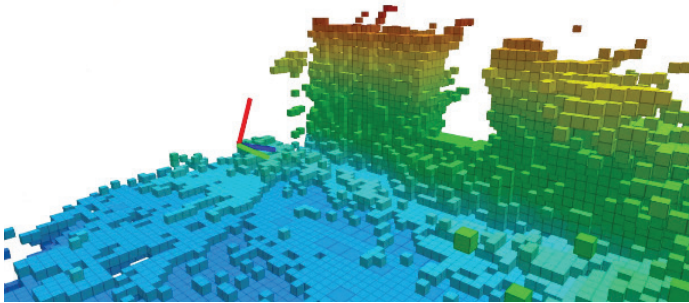
Figure 12. OctoMap representation of the interior of a building generated by the Skybotix UAS (source: ICARUS consortium).

4. Conclusions

This chapter discussed the ICARUS unmanned aerial systems that were developed in order to assist human search and rescue workers, by providing them aerial systems which can fulfil multiple roles: act as an eye in the sky, perform large-area or small-area mapping and surveillance operations, detect remaining survivors outside, perform detailed 3D reconstructions for

damage assessment and structural inspection, act as wireless repeaters, search for victims inside semi-demolished buildings, drop rescue kits and floatation devices, … As it is impossible to fulfil all these roles with one single aircraft, three main platforms were developed: a fixed wing endurance aircraft, which can stay airborne for multiple days and which even set the world record (81 hours) for this aspect [2], an outdoor rotorcraft which performs more targeted operations at lower altitudes and an indoor rotorcraft which is small, agile and intelligent enough to navigate in cluttered indoor environments.

Acknowledgements

Author details

Rudin Konrad[1]*, Daniel Serrano[2] and Pascal Strupler[3]

*Address all correspondence to: konrad.rudin@mavt.ethz.ch

1 Swiss Federal Institute of Technology in Zurich, Zürich, Switzerland

2 Eurecat, Barcelona, Spain

3 Skybotix AG, Zürich, Switzerland

References

[1] Nikolic J, Rehder J, Burri M, Gohl P, Leutenegger S, Furgale PT, Siegwart R. A synchronized visual-inertial sensor system with FPGA. In: IEEE, editor. IEEE International Conference on Robotics and Automation (ICRA). Hong Kong, China; 2014

[2] Oettershagen P, Melzer A, Mantel T, Rudin K, Lotz R, Siebenmann D, Siegwart R. A solar-powered hand-launchable UAV for low-altitude multi-day continuous flight. In: IEEE, editor. IEEE International Conference on Robotics and Automation. Seattle, Washington USA; 2015

[3] Pixhawk. Pixhawk [Internet]. Available from: https://pixhawk.org/

[4] Leutenegger S, Melzer A, Alexis K, Siegwart R. Robust state estimation for small unmanned airplanes. In: IEEE, editor. IEEE Conference on Control Automations (CCA). Juan Les Antibes, France; 2014

[5] Serrano D. Introduction to JAUS for Unmanned Systems Interoperability. NATO Science & Technology Organization – STO-EN-SCI-271 Report. 2015

[6] Bircher A, Alexis K, Burri M, Oettershagen P, Omari S, Mantel T, Siegwart R. Structural inspection path planning via iterative viewpoint resampling with application to aerial robotics. In: IEEE, editor. IEEE International Conference on Robotics and Automation (ICRA). Seattle, Washington USA: IEEE; 2015

[7] Vempati AS, Agamennoni G, Stastny T, Siegwart R. Victim detection from a fixed-wing uav: Experimental results. In: Advances in Visual Computing. Las Vegas, USA: Springer; 2015

[8] Serrano D, Uijt de Haag M, Dill E, Vilardaga S, Duan P. Seamless indoor-outdoor navigation for unmanned multi-sensor aerial platforms. In: The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, Volume XL-3/W1, 2014

[9] Omari S, Gohl P, Burri M, Siegwart R. Visual industrial inspection using aerial robots. In: IEEE, editor. 3rd International Conference on Applied Robotics for the Power Industry (CARPI). Foz do Iguassu, Brazil; 2014

Unmanned Ground Robots for Rescue Tasks

Karsten Berns, Atabak Nezhadfard, Massimo Tosa,

Haris Balta and Geert De Cubber

Additional information is available at the end of the chapter

Abstract

This chapter describes two unmanned ground vehicles that can help search and rescue teams in their difficult, but life-saving tasks. These robotic assets have been developed within the framework of the European project ICARUS. The large unmanned ground vehicle is intended to be a mobile base station. It is equipped with a powerful manipulator arm and can be used for debris removal, shoring operations, and remote structural operations (cutting, welding, hammering, etc.) on very rough terrain. The smaller unmanned ground vehicle is also equipped with an array of sensors, enabling it to search for victims inside semi-destroyed buildings. Working together with each other and the human search and rescue workers, these robotic assets form a powerful team, increasing the effectiveness of search and rescue operations, as proven by operational validation tests in collaboration with end users.

Keywords: unmanned ground vehicles, search and rescue

1. Introduction

1.1. Motivation

Ground robots have demonstrated to be a useful tool when dealing with post-disasters intervention and in particular for Urban Search and Rescue (USAR).

From the experience of the search and rescue (SAR) operators that have been consulted, it came out as the best ground robot for SAR operations would have such large variety of capabilities and tools that it would not be feasible to merge in a single machine. Hence, the strategy to use two different robots with well-defined and complementary capabilities:

- A large unmanned ground vehicle (UGV) to perform heavy duty jobs and collect information about dangerous places

- A small UGV to enter tight places, look for victims, and provide indoor view for danger assessment

These two robots can be either used together within one SAR team to have the most effectiveness or can be used independently of each other.

The advantages of using UGVs in disaster scenarios are multiple:

- Faster location of victims

- Shorter rescue times

- Less risk for the SAR operators

- Faster assessment of damage to buildings

The disadvantage to have two machines to develop instead of a single one can be partially compensated using similar structures and control programs on both robots [1].

1.2. State of the art

Kleiner [2] addresses the problems of robot localization, environment mapping, team coordination, and victim detection. In particular, an RFID-SLAM approach is used to close the loop when mapping. Robot and human's position is tracked respectively by slippage-sensitive odometry and pedestrian dead reckoning (PDR). Maps are then used for both centralized and decentralized coordination of rescue teams. Data collected by robots are available to SAR operators through wearable devices like head-mounted display (HMD).

Michael [3] combines maps generated by an unmanned aerial vehicle (UAV) and a ground robot used as a moving base. The cooperation between UGV and UAV in particular is addressed. The purpose is to explore compromised nuclear power plants that are too risky for humans due to the high level of radioactivity. Maps are generated from 3D sensor data using the Iterative Closest Point (ICP) approach. Maps are then corrected based on the odometry readings.

Murphy focuses [4] on robots for underground mine rescue. The physical challenges emerging from different scenarios are addressed and different platforms are proposed to better fit the various scenarios. In particular, most of the platforms proposed are mobile robots equipped with tracks and a dexterous manipulator and they are teleoperated through fibre-optic cable.

In the Viewfinder Project [5], robotic tools were developed for disaster management and for supporting fire-fighting services. However, the project concentrated mostly on developing the teleoperation and autonomous navigation capabilities [6, 7] and did not consider the mobility of the unmanned vehicles on rough terrain. This dichotomy is often seen in research projects: either they concentrate on conducting research on robot mobility in rough terrain or either on increasing the cognitive/intelligent behaviour of the robotic assets, but seldom on the combination of both research domains.

In the EU-funded project NIFTi, a system for supporting human-robot teams during disaster response has been designed and developed [21]. A UGV is used in different scenarios to explore the disaster area and look for victims. In a similar way as the ICARUS SUGV, the NIFTi UGV has some capabilities necessary to accomplish the USAR missions:

- Automatic victim detection

- Creation of metrical map based on LIDAR

- Path planning

- Multimodal human-robot interaction

Furthermore, the NIFTi UGV has some higher-level features that make it able to reason and infer more complex concepts about the environment.

The localization is performed fusing visual and inertial odometry and correcting the result with an estimate obtained through the ICP algorithm.

The high level representation of the sensory inputs is addressed using a topological representation of the environment consisting in a graph whose nodes can be either relevant objects or regions obtained by segmentation of the metric map. The robot uses this representation to autonomously plan a method to execute a task.

The EU-funded project TRADR is a sequel to the aforementioned project NIFTi [22]. After the earthquake in Amatrice, Italy, deployment with a teleoperated UGV and some UAVs has been performed. The UGV carries similar sensors as the ICARUS SUGV, with the exception that the former has a LadyBug3 omnidirectional camera while on the SUGV it has been preferred to mount single cameras in the most critical points. The laser scanner is almost the same model, but in the TRADR UGV, it is mounted on the front on a rotating unit, to provide a 3D point cloud, while on the SUGV, the two laser scanners are fixed on the sides and the 3D point cloud is provided by a time-of-flight camera.

1.3. Subtasks

Breaking with the tradition of focusing on only one research domain, the ICARUS project developed intelligent robotic agents with high mobility on rough terrain [1]. The UGVs represent, within the ICARUS project, the core of the assistive robots during land disasters like earthquakes, floods, landslides, etc. The UGVs are not meant to be a substitute of human search and rescue operators but they are instead a complementary tool to assist them and extend their operational possibilities.

The large unmanned ground vehicle (LUGV) is intended to be primarily a means to open the way to rescuers whenever the way is obstructed by debris. The possibility to mount the jackhammer makes it useful to break with a certain speed walls and concrete slabs. Using the gripper then makes it possible to remove debris and stones that obstruct the entrance to a damaged building. Whenever the structure of the building is not completely stable, it is possible to use the gripper to place some struts to stabilize it; in this way, the risk for the rescuers is minimal as they can perform this operation remotely.

When the entrance to the collapsed building is large enough to allow the passage of the small unmanned ground vehicle (SUGV), then its box is hooked to the end-effector of LUGV arm and it is deployed. If the ground floor is still not accessible, it is possible to deploy SUGV on a balcony or a roof up to 3 m.

The SUGV can explore the inside of a building, teleoperated from a remote area. The operator has a view of the environment around the robot, thanks to the numerous sensors and the mapping system. Knowledge of the obstacles that are near the robot is necessary when doing manoeuvres in tight space in order to avoid collisions that could make the robot unusable.

With an infrared camera, SUGV is able to find heat traces belonging to victims that cannot move. When a victim is found then the operator can annotate its position and deliver a small rescue kit like water or oxygen. When the communication bandwidth is enough, bidirectional voice communication with the victim is possible in order to give instructions and to reassure the victim.

2. Mechatronics of the large unmanned ground vehicle

2.1. Design concept

The LUGV (**Figure 1**) was originally a commercial vehicle built by the French company Metalliance [8]. It was adapted to the purposes of ICARUS project and provided to the University of Kaiserslautern. Its specifications are shown in **Table 1**. The main power is provided by a diesel engine which in turn actuates a hydraulic pump. This latter pumps oil into the pistons that actuate the two tracks. A turbine attached to a generator is actuated by the oil flow as well and this latter provides 220 V AC.

LUGV has a high mobility on uneven terrain due to its big caterpillars that allow in-place rotations as well. A 5 degree of freedom hydraulically actuated manipulator is mounted on top of the vehicle. Different tools can be attached to the end-effector, a gripper, a jackhammer, or a box to transport the SUGV.

Figure 1. Large unmanned ground vehicle (Source: ICARUS).

Dimensions	3 m × 2 m × 1.8 m
Weight	4 t
Payload weight capacity	300 kg
Power capacity	1 kW at 12 V DC
Maximum speed	18 km/h
Autonomy	3 h
Operational range	2500 m
Computing power	Intel Core i7-3610, 4GB RAM
Control software	FINROC (C++)
Safety mechanism	4 emergency stop buttons, 1 wireless passive stop system
Main power	Diesel engine
Locomotion system	6 wheels inflated with polyurethane foam covered by two caterpillars
Minimum turning radius	0 m

Table 1. LUGV specifications.

The control program runs on a PC that communicates with a PLC, this latter deals directly with the low level hardware. The manipulator has its own controllers that are connected to the main PC.

2.2. Sensor system

The position and orientation on LUGV are provided respectively by a global navigation satellite system (GNSS) and an inertial measurement unit (IMU). As there is quite some space on the LUGV, it has been possible to mount two GNSS antennas, making the position measurement more precise.

A stereo-camera system was mounted on the front section bar tower, with the aim of providing a dense 3D point cloud used for mapping and obstacle detection [9, 10]. The choice of the stereo-camera system is justified by a bigger range with respect to a time-of-flight camera. One camera of the stereo-system is used also as direct visual feedback by the remote operator. Since this camera is attached to the main structure of the robot, an IP camera is mounted on the arm next to the elbow joint, which provides visibility about the arm in order to not hit any object during manipulation activities. Finally, a USB webcam was installed on the end-effector to have direct visibility on the tool.

Two laser range finders are mounted on the front and the rear side for obstacle detection, in a similar fashion as on SUGV but the sensor type on LUGV has a bigger range and a smaller resolution; further details are shown in Section 6.1.

2.3. Mechatronics of the arm

The LUGV manipulator is a 5 degree of freedom arm, which is hydraulically actuated. The workspace of the manipulator is shown in **Figure 2**.
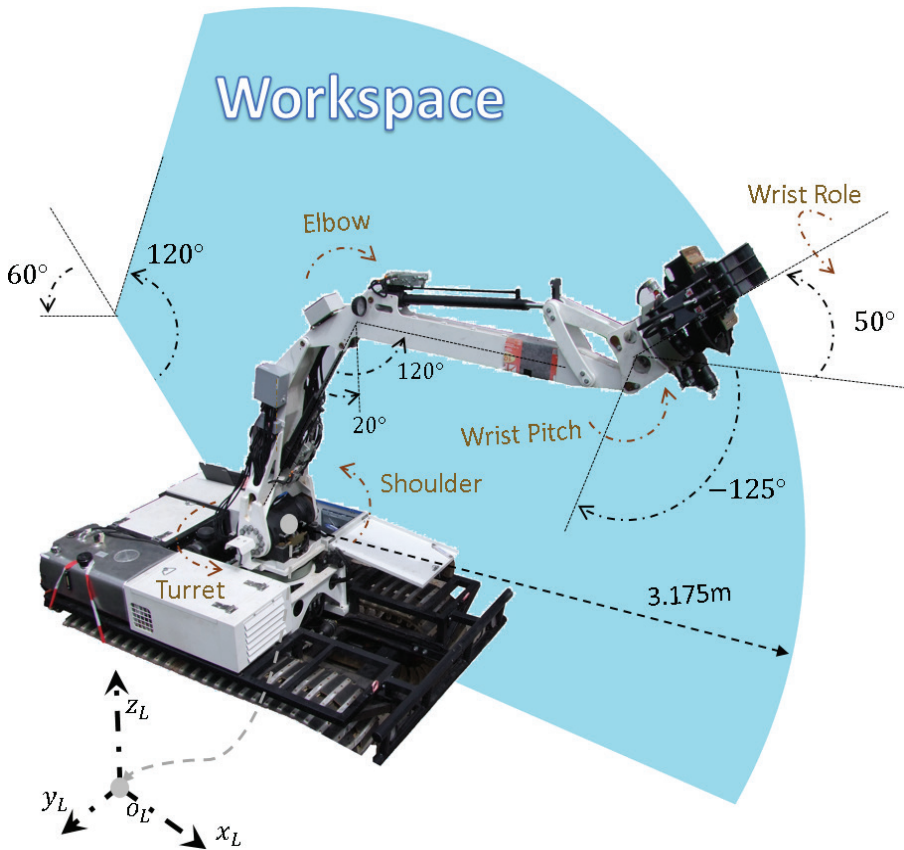
Figure 2. Workspace for the manipulator arm on the LUGV (Source: ICARUS).

A simple PID controller is used to control the joint angles. During operation, the temperature and pressure of the hydraulic oil is changing and therefore the internal dynamics of the hydraulic actuator is time variant. This can result in undesirable oscillations if the PID controller is set to high gains. The issue of stability can be alleviated by reducing the PID gains and consequently decreasing the precision. Due to the safety issues of such a large vehicle, the PID controller has to be chosen in low gain to guarantee stability. We adopted a different methodology to overcome the problem of stability. In order to increase the speed of operation and precision, a virtual trajectory was calculated to force the arm towards the desired trajectory; nevertheless, this method had no benefit in remote control scenario where the user directly moves the manipulator. Our experiments proved that for convenient operation of the arm a haptic controller with force feedback is necessary to safely control such a highly dynamic arm.

The LUGV manipulator was equipped with three different tools. The gripper tool is used for grabbing and moving obstacles such as metal sheets, bricks, etc. The controller provides satisfactory accuracy for grasping task. The other tool is jackhammer. This tool is used to breach the concrete walls and in several demonstrations proved to be a useful way for the SUGV to enter hazardous buildings. SUGV box is another tool that is used to carry and deploy SUGV near the rescue area. Deploying SUGV is vital to the rescue operation since SUGV can cover a maximum radius of 50 m in a reasonable time. Where LUGV can approach the building, it can save the battery power of the SUGV for more indoor operation. LUGV also has the capability of deploying SUGV up to 3 m above the ground, and this provides flexibility to the rescue operation where SUGV can be directly placed on the first floor of a building.

3. Mechatronics of the small unmanned ground vehicle

3.1. Design concept

The SUGV (**Figure 3**) is a commercial robot originally produced by the British company Allen Vanguard as a teleoperated robot for bomb defusing [11]. The specifications of the system are shown in **Table 2**.

The platform motion is provided by two tracks actuated by an electric motor with a gearbox. The tracks are a bit higher in the front side, and this permits to climb steps up to 20 cm.

The 5 degree of freedom manipulator allows to open doors, grasp small objects, or extend the visual feedback where the platform cannot arrive, like through small holes or above little walls. Each joint of the arm is actuated by a DC motor and a worm gearbox.

The low-level control of the caterpillar motors and the manipulator is performed by a dedicated motor controller and a digital signal processor (DSP), respectively. The DSP is in charge also to control the lights and gather information from the joint encoders. The main processing
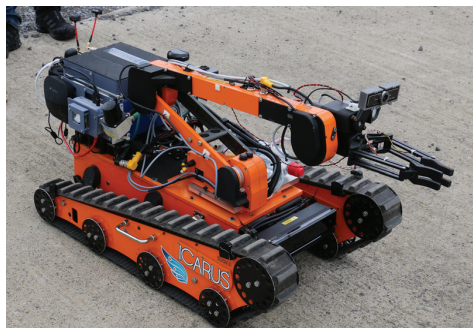
Figure 3. Small unmanned ground vehicle (Source: ICARUS).

Dimensions	104 cm × 53 cm × 56 cm (with closed arm)
Weight	60 kg
Main power	2 × 24 V DC LiPo batteries
Propulsion	2 caterpillars actuated by DC motor + gearbox
Arm payload	20 kg
Autonomy	2 h
Maximum speed	2 km/h
Computing power	Intel Core i7-3610, 4GB RAM
Control software	FINROC (C++)
Safety mechanisms	Wireless controller, emergency button
Minimum turning radius	0 m

Table 2. SUGV specifications.

power is provided by a PC mounted on the rear part of the chassis. It is present as a second processing unit in an Odroid, which is used only to collect data from some sensors.

3.2. Sensor system

The SUGV was provided with a rich sensor configuration for the purpose of ICARUS:

- An IMU and GPS provide basic telemetry like position and orientation.

- A Kinect of the second generation is installed in the front part of the chassis. It provides a dense 3D point cloud used for navigation and obstacle avoidance. Two laser range finders (LRF) are mounted on both sides for the same purpose. Two different maps are built from the Kinect and LRF and then merged together afterwards (Section 6.1).

- An RGB camera is integrated into the Kinect, while a second camera is installed on the gripper and is used when it is needed to grasp objects with the manipulator or when it is necessary to extend the visual feedback to places that are inaccessible for the whole robot, like below furniture, or inside small holes. A third RGB camera is mounted on the rear side. It is set to a low resolution to reduce the network load. This camera is used mainly when doing manoeuvres in tight space where rear vision is required.

- An infrared camera with a quantum cascade detector [12] were mounted after the demonstrations to find the heat trace of alive victims as well as the presence of CO_2 in the air.

- As requested by the end-users, a CO_2 sensor was installed to measure CO_2 in the air with direct contact.

- A microphone and speaker were mounted as well to allow a direct communication with victims when possible or text-based messages when the communication bandwidth is low.

3.3. Mechatronics of the arm

The SUGV manipulator (**Figure 4**) has three rotational joints on the arm and two rotational joints on the wrist. It is able to perform the required tasks inside its workspace. The arm is designed to perform low speed high precision object manipulation, where its actuators are equipped with worm drive gear-boxes to maximize the load capability of the arm and reduce the negative effect of the load disturbance on position control. Henceforth, the arm is capable to lift up to 20 kg weight in compact mode and 8 kg in fully stretched mode.

The joints are using magnetic absolute encoders that are resistant to dust and moisture with a resolution of 0.1 degree. The sensory feedback provides high precision position control which is only limited with the backlash of the drive boxes. This feedback can also be applied to provide force feedback for the haptic device. Our experiment demonstrated that the force feedback calculated based on position error can help the user to have a better understanding of the motion and adapt itself to the slow speed of the manipulator. The force feedback can
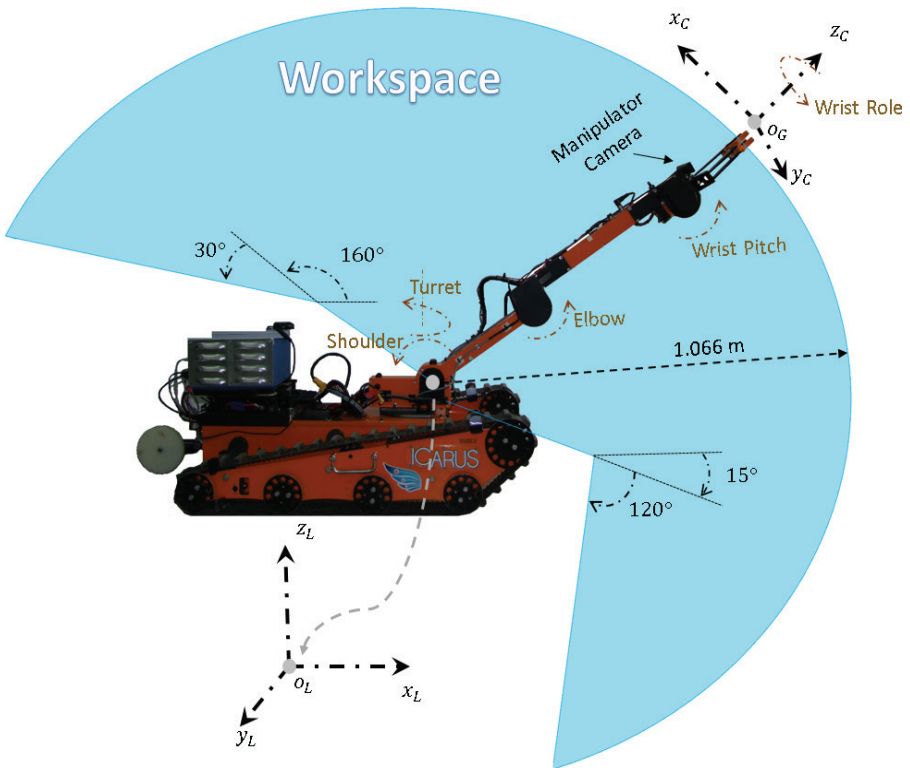

Figure 4. The workspace of the SUGV manipulator (Source: ICARUS).

be influential in teleoperation with high vision delay. In teleoperation tasks, especially in disaster area, the communication bandwidth is low and the video quality is not good. In such a situation, the force feedback could be of great value since it provides the user fast response from environment preferred to the delayed camera image.

4. Control concept

On both robots, FINROC (Framework for INtelligent RObot Control) is installed, a C++/Java framework [13] developed by RRLab explicitly to control robots in an easy and flexible way.

The modularity of FINROC allows the user to add modules that perform different tasks and communicate with each other through ports. The user does not have to care about details like scheduling, data exchange, or multithreading, as they are managed in the background by the framework. The data type that can be passed through ports can be various, such as values, images, point clouds, etc. The basic requirement is that data must be serializable to be passed.

Two Java-based tools can be used to monitor the program: Finstruct and Fingui. The first shows a tree with the modules that are running within the program and allows the user to connect the ports manually in runtime without the need to recompile; this feature is particularly useful when doing tests on the field and it is required to do fast changes to the program. Fingui is a graphical interface that allows to visualize the output of some ports and to give inputs to others using some predefined graphic widgets.

The structure of the software running on our UGVs has at the top level the graphical interface or the Joypad; both can be used by the operator to provide control inputs (**Figure 5**).

Such controls are then processed by the control program (IcarusControl) and converted to hardware commands. The control program is in charge of the high level processing of data coming from the sensors. The mapping, localization, navigation, and obstacle avoidance are tasks performed by modules of this layer.

The motion commands for both platform and manipulator are sent by the Hardware Abstraction Layer (HAL) to the related hardware. The HAL is responsible also for collecting data from hardware peripherals such as manipulator encoders and send it to the control program.

Since FINROC is not a ROS node, it could not be directly connected to the ICARUS Command and Control interface (C2I) [14], which is ROS-based. For this purpose, an interface was created using the JAUS (Joint Architecture for Unmanned Systems) library [15].

On the robot, the interface is a FINROC module that acts as a bidirectional gate: on one side, it receives commands from the C2I to be sent to the robots. On the other side, the interface receives sensor data from the robot and sends them to the C2I. The available data that are sent through the interface in both directions is shown in **Table 3**.
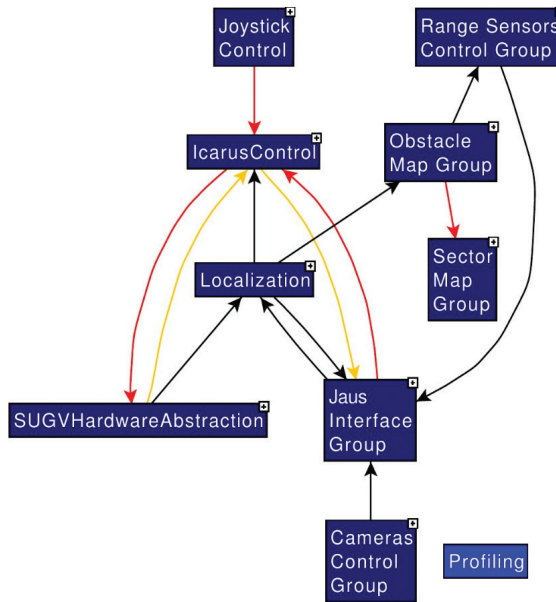
Figure 5. Control structure (Source: ICARUS).

	SUGV	LUGV
C2I to UGV		
End-effector desired Cartesian position	V	V
Joints desired position	V	V
Gripper control	V	V
Light switch	V	V
Engine switch		V
Manipulator enable	V	V
Platform motion	V	V
Waypoints list	V	V
Tool change selector		V
Tool lock switch		V
Enable audio communication	V	
Enable speech over text	V	
Reset localization	V	V
Text	V	
Digital video sensor	V	

	SUGV	LUGV
UGV to C2I		
Roll, Pitch, Yaw	V	V
GNSS	V	V
RGB images	V	V
Point cloud	V	V
Battery voltage	V	
End-effector actual Cartesian position	V	V
Joints actual position	V	V
Low fuel alarm		V
Text	V	
Victim waypoint	V	
CO_2 value	V	

Table 3. Data sent through the interface.

An asynchronous call-back function is called whenever a message is received by the C2I. Depending on the type of the message that is received, the call-back converts it to the proper FINROC-compatible data format and forwards the data to the main control program. At every cycle of the frame the interface gets the input data from the robot and after proper conversion to JAUS formats calls the relative function to forward the data to the JAUS layer.

5. Simulation

Simulation has been a valuable tool to develop both robots, as shown in **Figures 6** and **7**. Because of the difficulty and the costs, both in time and money, to test the platforms (the LUGV in particular), some simulations were created using different simulation engines.

The last and most accurate simulation performed makes use of the Virtual-Robot Experimentation Platform (V-REP) [16].

A CAD model of both robots has been drawn, part-by-part, and then assembled in the simulation and provided with dynamic properties measured on the real robots, such as mass, moment of inertia, centre of mass, and friction coefficient.

In the simulation, the robot's hardware is replaced by a physical model of the real platforms. The higher level control program used to control the simulated robots is the same that is used to control the real robots.

Simulation has been particularly useful to test the collision avoidance system, for which a big space with many different obstacles was necessary.
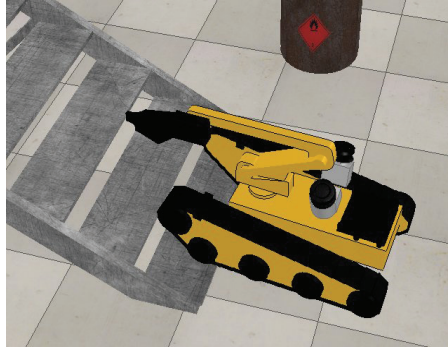
Figure 6. SUGV simulation (Source: ICARUS).
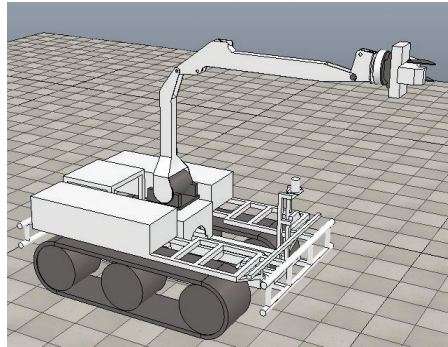
Figure 7. LUGV simulation (Source: ICARUS).

6. Navigation of the UGVs in an unstructured environment

6.1. Mapping

A proper mapping system is available on both robots. It processes the information about the environment gathered by the sensors and it converts to a format that is usable by the collision avoidance system and by the human operator. This system is composed of the following parts:

- Laser range finders (LRF) + stereo-camera (LUGV only)

- Grid map

- Sector map

On the LUGV, two LRFs are installed, one in the front and one in the back. Both are attached to a section bar structure fixed directly to the bumpers. The sensor is a SICk LMS511, a planar laser scanner protected by a case against rain and dust. The scan has a horizontal field of view (FOV) of 180°, the scan range is about 80 m, for a minimum angular resolution of 1/6°. Sensor frequency is set to 10 Hz, which means that 10 scans per each device are done in a second and each scan provides 1080 planar points. The height of the sensors is about one-half of the track height, so that if an obstacle intersects the scan plane it is considered not traversable, everything that lies below the scan plane is considered traversable. Both laser scanners are not completely outside the vehicle but they are protected by its bumpers in such a way that if the obstacle detection fails and an obstacle is hit, the first part to hit is the bumper instead of the more delicate and expensive laser scanner. Anyway a suitable position is found considering also the needing to have the least possible parts belonging to the vehicle that obstruct the FOV of the sensor.

On the SUGV, two LRFs are installed as well. Smaller than the ones mounted on LUGV, they are located on both sides, with the plane of scan just above the tracks. Even if the FOV of this scanner type is 270°, they are limited to 180° to avoid that parts of the robot itself fall within the FOV. The scan range is 10 m and the minimum angular resolution is 1/3°.

At every cycle both scans from the two devices are merged together and converted from the sensor 2D reference frame to the vehicle 3D reference frame, in this way a 3D point cloud is obtained.

A traversability map [17] is built out of the point cloud. This map is a 2D grid of cells built around the vehicle, and sizes are by default 20 m × 20 m with a resolution of 0.25 m for LUGV and 6 m × 6 m with a resolution of 0.05 m for SUGV.

In the pre-processing phase, the point cloud is filtered and the points too close to the sensor are removed because they are assumed to belong to the robot itself. The point cloud is then filled into the grid map. For each point belonging to the cloud, the corresponding cell in the map is found. Each cell keeps a count of how many points lie inside it.

To consider a cell as an obstacle, it is necessary that the number of points contained is more than a specific threshold. Such threshold is not constant for all cells but it is inversely proportional to the radial distance from the vehicle; this is done to keep into account that the point density decreases in far cells.

A cell can have on of the following labels:

• UNKNOWN: if the cell is out of the FOV of the sensors and it was never explored

• FREE: if it is within the FOV of the sensors and the number of points inside is lower than the threshold

• EXPECTED FREE: if it was FREE in the previous cycle and it is now out of the FOV

• OBSTACLE: if the number of points inside is higher than the threshold

• EXPECTED OBSTACLE: if it was OBSTACLE in the previous cycle and it is now out of the FOV

• ROBOT: are the cells covered by the robot

The meaning of EXPECTED FREE and EXPECTED OBSTACLE is that we have some information about these cells from the previous cycles but they are currently not covered by the sensors, so we expect that the previous information is still valid but we are not sure.

The grid map is refreshed every cycle and it is connected with the vehicle odometry from the inertial measurement unit (IMU). When the robot moves, the map is shifted of an amount depending on the distance travelled by the robot. In this way, the robot is always in the centre of the map and the obstacles shift. If an obstacle goes out of the map boundary its information is lost. If the robot rotates then the virtual robot in the map rotates while the obstacles around are fixed.

The sector map is the next level of the obstacle detection system. It is a local map built out of the grid map. It contains information about the very close distance from the robot.

Two types of maps are used: a Cartesian and a polar map. A total of 18 maps belonging to the two types are used. Each map contains 10 sectors and each sector stores the minimum free range. The purpose of the sector is to know how much manoeuvrability has the robot in order to not collide with any obstacle.

Sector maps are filled from cells of the grid map. In particular, for each OBSTACLE and EXPECTED OBSTACLE cell the distance from each sector origin is performed and if such distance is less than the sector maximum range then the current sector range is set equal to the cell distance. To speed up the processing, only the cells that lie within the FOV of the sensors are computed.

Sector maps are then used to feed the behaviour-based collision avoidance network [18] that is based on this information and on the motion control from operator decides which movement to perform.

Figures 8–10 depict the different stages of the mapping process: evolving from a point cloud to a grid map to end up with a sector map.

On the SUGV, a further mapping is performed based on a dense 3D point cloud that is gathered by the Kinect. The same process is applied as for the LRFs and the behaviour-based collision avoidance system is in charge to fuse the information from both sector maps (Kinect and LRFs).
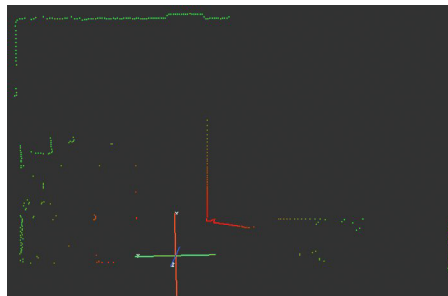
Figure 8. 2D view of 3D point cloud (Source: ICARUS).
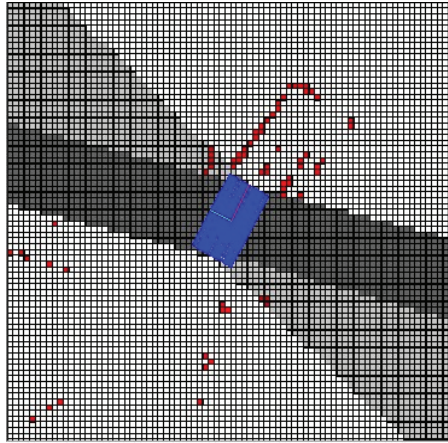
Figure 9. Grid map (Source: ICARUS).
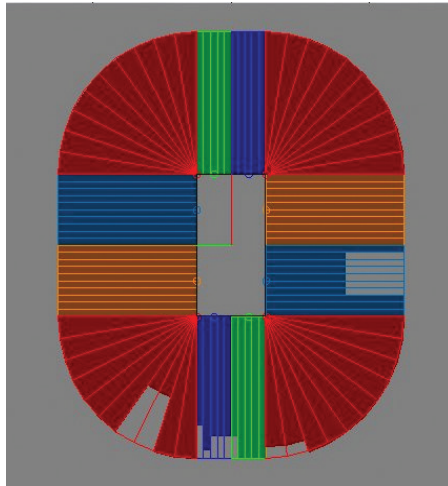
Figure 10. Sector map (Source: ICARUS).

6.2. Path planning

The navigation system, further discussed in [19], is the same on both robots, except some parameters that are strictly platform-dependent. Since both robots have the capability to perform in-place turns, it was preferred to perform a simple point-approach navigator instead of a more complex trajectory tracking.

The difference is that with the point-approach technique a sequence of geographical points must be provided by the user. Starting from where the robot is located as soon as it receives the sequence, it turns in-place towards the first point in the list and it begins moving in a virtual straight line to reach it. When the point is reached, the robot goes on to reach the second point in the list and so on.

The trajectory between two consecutive points is subordinate to the collision avoidance system. That means that the robot tries to move in a straight line but if there are obstacles on its trajectory it performs manoeuvres in order to avoid them and it re-plans a new trajectory to the next point.

To have a correct navigation it is obvious that a precise localization system is correct, in both indoor and outdoor environments.

7. Control of the manipulator arm

The control loop of the manipulator is illustrated in the **Figure 11**. The lower level of the controller is a digital signal processing (DSP) unit, responsible to calculate and acquire the sensor data to control the position of the joints. It also uses the CAN Bus interface to send the sensory data to the onboard computer where the higher level calculations such as inverse and forward kinematics are performed. The electronics are designed very flexible since many different
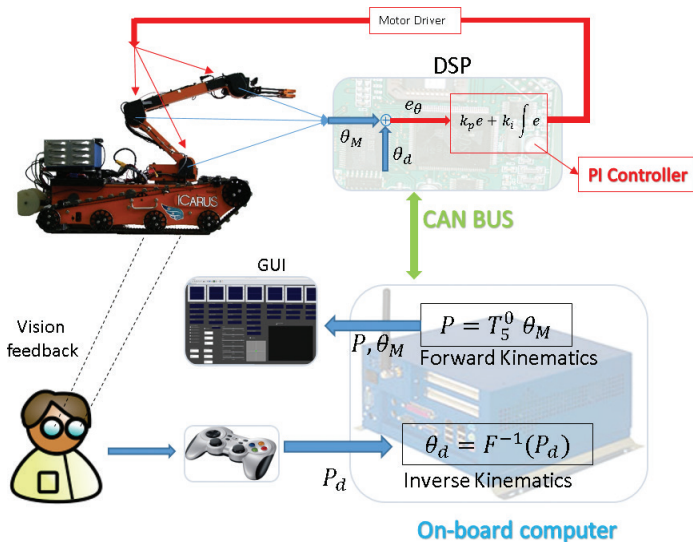
Figure 11. The control loop of the manipulator arm (Source: ICARUS).

features and I/O interfaces can be added to the system later on. For instance, touch sensors are added to the gripper later on to give the force feedback of the contact force. Also, laser pointer with circular pattern was added to give better sense of distance to the user. Different lights and analogue sensors could be extended due to versatile design of the electronics.

The onboard computer calculates the kinematics and assists the user in control. Based on the experiments, four different control modes are coded for user interface. In workspace mode, the user is able to control the velocity of the manipulator with respect to robot frame. This mode is useful when the user has a direct eye contact with the robot, without using an onboard camera. The configuration mode gives the user possibility to control each joint. This mode is for test and calibration of the manipulator and is not recommended for manipulation task. In the camera mode, the user is controlling the speed of the tool centre point (TCP) with respect to camera frame placed on the end effector. The last mode is Pose mode, where instead of velocity, directly the desired workspace position and orientation (P_d) of the TCP is commanded by the user. This mode is particularly useful when haptic joysticks or an exoskeleton is used to control the arm. Using this mode, the position and orientation of the user's hand will be directly mapped to the pose of the manipulator. This makes the manipulation much easier and faster; indeed, complicated tasks without using haptic devices are almost impossible to perform with articulated arm.

8. Experimental validation

Operational validation trials were performed in Marche en Famenne, Belgium, in order to assess the capabilities of the SUGV and LUGV against the user requirements. On the LUGV, the 3D vision sensor [20] was integrated into the robotic control framework FINROC. After mounting the sensor on the vehicle, a calibration of the sensor (see **Figure 12**) has been carried out to synchronize the sensor coordinate frame with the robot coordinate frame. Several tests to ensure that the detection of obstacles, the traversability information, and the visual feedback were correctly set up.

Subsequently, the autonomous navigation was evaluated: the vehicle has been given a list of GPS waypoints and it was successfully able to reach them without any direct control. Also, the manipulation capabilities of the LUGV were tested. First, the box with the SUGV inside has been lifted to a height of more than 2 m and the SUGV has been deployed on the roof of a test building (see **Figure 13**).

Before entering the building, a door had to be opened. This was accomplished using the feedback-controlled manipulator arm of the SUGV. Then the vehicle had been driven remotely inside the damaged building using its own lights and a camera to look for victims in all rooms, passing through a hole in the wall and descending stairs (see **Figure 14**).

All the available tools have been successfully mounted on LUGV: the box for the SUGV, the gripper, and the jackhammer (see **Figure 15**). Then some rocks have been grasped with the gripper and a concrete piece has been broken to assess the capabilities of the jackhammer (see **Figure 16**).

Afterward, the small ground vehicle was deployed together with an indoor multi-copter to explore another building.
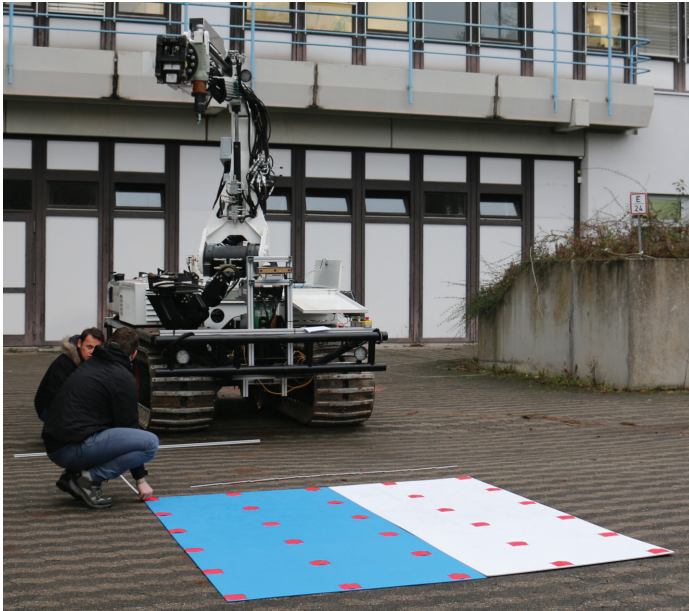
Figure 12. Calibration of the 3D vision sensor. The sensor can be seen mounted on the sensor tower of the LUGV (Source: ICARUS).

Figure 13. The LUGV uses its arm to lift the transport box containing the SUGV (Source: ICARUS).

Figure 14. Due to its small size, the SUGV is able to drive through narrow openings (Source: ICARUS).
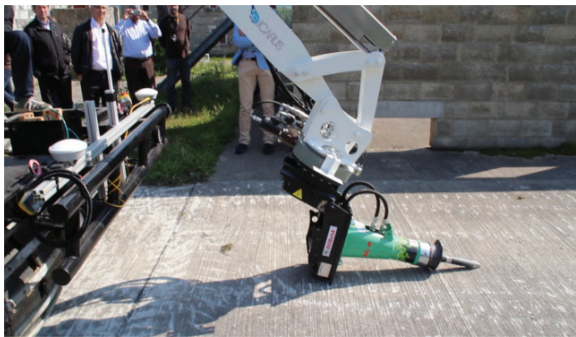
Figure 15. The jackhammer is being attached to the arm of the LUGV (Source: ICARUS).

Figure 16. The LUGV uses its jackhammer to breach through a concrete plate (Source: ICARUS).

A posteriori, the human victim detection sensor was integrated on the SUGV (**Figure 17**). It was mounted next to the arm as this fulfils most of the requirements on the degree of freedom of the manipulator arm, field of view for the human detection sensor, and avoidance of collision-prone configurations.
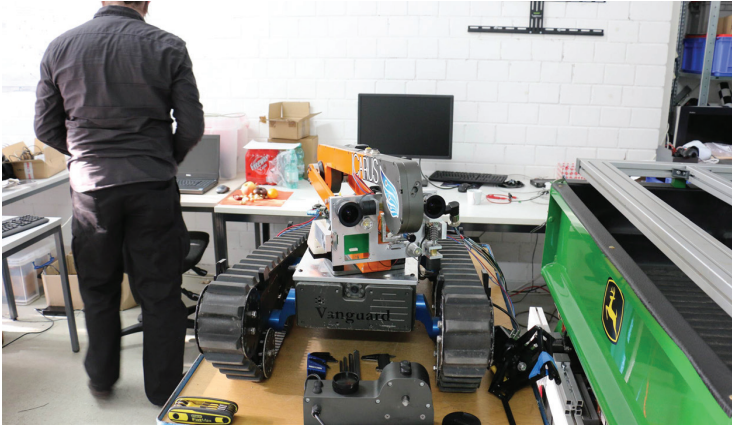
Figure 17. The SUGV with the human detection sensor mounted in front of the manipulator arm (Source: ICARUS).

9. Main issues and considerations

Various experiments conducted for validation purposes have shown that both UGVs can still be used successfully for search and rescue tasks. The formal validation showed that both the SUGV and LUGV fulfil most of the requirements that have been set out by end users. However, some requirements are currently also not fulfilled and it is important to learn from this. Aspects where more research is required are:

- Compromising system complexity and robustness in large systems, as both systems sometimes had robustness issues.

- Precise manipulation with powerful hydraulic arms, as was not possible to achieve the desired precision with the hydraulic actuator.

- Vibration isolation of sensors, or rendering sensors less prone to noise induced by vibrations.

- Stair climbing of heavily packed robot systems, as the SUGV is capable of climbing stairs, but with the full sensor load, this becomes very difficult for steep stairs designs. A more suitable platform for this task would be equipped with orientable flippers, as it has been done in the EU-funded projects NIFTi and TRADR [21, 22].

- Long-distance, non–line-of-sight communication for operations inside buildings, as we did still lose communication to the SUGV from time to time.

10. Conclusions

In this chapter, the ground robots employed in the ICARUS project were described. The LUGV is a heavy-duty supporting machine used to free the way from debris and open a safe passage to the SAR operators. The SUGV is a versatile robot used to explore and search victims within damaged buildings. Both platforms have been described in detail and their advantages and drawbacks related to SAR missions have been addressed. Furthermore, the SUGV has been compared with similar robots used in other projects.

Acknowledgements

Author details

Karsten Berns[1]*, Atabak Nezhadfard[1], Massimo Tosa[1], Haris Balta[2] and Geert De Cubber[2]

*Address all correspondence to: berns@cs.uni-kl.de

1 Technische Univeristät Kaiserslautern, Kaiserslautern, Germany

2 Department of Mechanical Engineering, Royal Military Academy of Belgium, Brussels, Belgium

References

[1] Armbrust C, De Cubber G, Berns K. ICARUS Control Systems for Search and Rescue Robots. Field and Assistive Robotics—Advances in Systems and Algorithms. Shaker Verlag, Aachen; 2014. pp. 1-16. DOI: ISBN-13: 978-3-8440-2753-2

[2] Kleiner A. Mapping and exploration for search and rescue with humans and mobile robots [dissertation]. Freiburg im Breisgau, Germany: Albert-Ludwigs-Universität Freiburg im Breisgau; 2007

[3] Michael N, Shen S, Mohta K, Mulgaonkar Y, Kumar V, Nagatani K, Okada Y, Kiribayashi S, Otake K, Yoshida K, Ohno K, Takeuchi E, Tadokoro S. Collaborative mapping of an earthquake-damaged building via ground and aerial robots. Journal of Field Robotics. 2012;**29**(5):832-841

[4] Murphy RR, Kravitz J, Stover SL, Shoureshi R. Mobile robot in mine rescue and recovery. IEEE Robotics and Automation Magazine. 2009;**16**(2):91-103

[5] Baudoin Y, Doroftei D, De Cubber G, Berrabah SA, Pinzon C, Warlet F, Gancet J, Motard E, Ilzkovitz M, Nalpantidis L, Gasteratos A. View-Finder: Robotics assistance to fire-fighting services and Crisis Management. In: IEEE, editor. IEEE International Workshop on Safety, Security & Rescue Robotics (SSRR 2009), 2009. Denver, CO; 2009. pp. 1-6. DOI: 10.1109/SSRR.2009.5424172

[6] Doroftei D, De Cubber G, Colon E, Baudoin Y. Behavior based control for an outdoor crisis management robot. In: Proceedings of the IARP International Workshop on Robotics for Risky Interventions and Environmental Surveillance; 2009. pp. 12-14

[7] De Cubber G, Doroftei D, Sahli H, Baudoin Y. Outdoor terrain traversability analysis for robot navigation using a time-of-flight camera. In: RGB-D Workshop on 3D Perception in Robotics; 2011

[8] Metalliance. Metalliance [Internet]. Available from: http://www.metalliance-tsi.com/en/ [Accessed: November 2016]

[9] De Cubber G, Doroftei D. Multimodal terrain analysis for an all-terrain crisis management robot. In: IARP HUDEM 2011. IARP; 2011

[10] Balta H, De Cubber G, Doroftei D, Baudoin Y, Sahli H. Terrain traversability analysis for off-road robots using time-of-flight 3d sensing. In: 7th IARP International Workshop on Robotics for Risky Environment-Extreme Robotics; Saint-Petersburg, Russia. IARP; 2013

[11] Allen Vanguard. Remotely Operated Vehicle-Digital Vanguard. Product Brochure; 2011

[12] Harrer A, Schwarz B, Gansch R, Reininger P, Detz H, Zederbauer T, Andrews AM, Schrenk W, Strasser G. Plasmonic lens enhanced mid-infrared quantum cascade detector. Applied Physics Letters. 2014;**105**(171112). DOI: http://dx.doi.org/10.1063/1.4901043

[13] Reichardt M, Föhst T, Berns K. On software quality-motivated design of a real-time framework for complex robot control systems. In: 7th International Workshop on Software Quality and Maintainability (SQM), in conjunction with the 17th European Conference on Software Maintenance and Reengineering (CSMR); March 2013; Genova, Italy. 2013

[14] Govindaraj S, Chintamani K, Gancet J, Letier P, Van Lierde B, Nevatia Y, De Cubber G, Serrano D, Bedkowski J, Armbrust C, Sanchez J, Coelho A, Palomares ME, Orbe I. The ICARUS project—command, control and intelligence (C2I). In: Safety, Security and Rescue Robots (SSRR); October 2013; Sweden. IEEE; 2013

[15] Serrano D. Introduction to JAUS for Unmanned Systems Interoperability. NATO Science & Technology Organization. STO-EN-SCI-271 Report. 2015

[16] Freese M, Singh S, Ozaki F, Matsuhira N. Virtual robot experimentation platform V-REP: A versatile 3D robot simulator. Simulation, Modeling, and Programming for Autonomous Robots. 2010;**6472**:51-62. DOI: 10.1007/978-3-642-17319-6_8

[17] Dargazany A, Berns K. Terrain traversability analysis using organized point cloud, superpixel surface normals-based segmentation and PCA-based classification. In: Workshop on Field and Assistive Robotics (WFAR); Lahore, Pakistan; 2014

[18] Kiekbusch L, Armbrust C, Berns K. Formal verification of behaviour networks including sensor failures. Robotics and Autonomous Systems. 2015;**74**:331-339

[19] Armbrust C. Design and verification of behaviour-based systems realising task sequences [dissertation]. München, Germany; Robotics Research Lab, Department of Computer Science, University of Kaiserslautern: Verlag Dr. Hut; 2015. p. 236. Available from: https://agrosy.cs.uni-kl.de/fileadmin/Literatur/Armbrust15a.pdf

[20] Vislab. 3DV-A. Product brochure; 2012.

[21] Kruijff GJM, Kruijff-Korbayová I, Keshavdas S, Larochelle B, Janíček M, Colas F, Liu M, Pomerleau F, Siegwart R, Neerincx MA, Looije R, Smets NJJM, Mioch T, van Diggelen J, Pirri F, Gianni M, Ferri F, Menna M, Worst R, Linder T, Tretyakov V, Surmann H, Svoboda T, Reinštein M, Zimmermann K, Petříček T, Hlaváč V. Designing, developing, and deploying systems to support human-robot teams in disaster response. Advanced Robotics. 2014;**28**(23):1547-1570. DOI: 10.1080/01691864.2014.985335

[22] Kruijff-Korbayová I, Freda L, Gianni M, Ntouskos V, Hlaváč V, Kubelka V, Zimmermann E, Surmann H, Dulic K, Rottner W, Gissi E. Deployment of ground and aerial robots in earthquake-struck amatrice in Italy (brief report). In: International Symposium on Safety, Security and Rescue Robotics; October 2016; Lausanne, Switzerland. IEEE; 2016

Unmanned Maritime Systems for Search and Rescue

Aníbal Matos, Eduardo Silva, José Almeida,
Alfredo Martins, Hugo Ferreira, Bruno Ferreira,
José Alves, André Dias, Stefano Fioravanti,
Daniele Bertin and Victor Lobo

Additional information is available at the end of the chapter

Abstract

The development of maritime unmanned tools for search and rescue operations is not a trivial task. A great part of maritime unmanned systems developed did not target such application, being more focused on environmental monitoring, surveillance or defence. In opposition to these applications, search and rescue operations need to take into account relevant issues such as the presence of people or other vessels on the water. Building upon user requirements and overall integrated components for assisted rescue and unmanned search operations (ICARUS) system architecture, this chapter addresses the development of unmanned maritime systems. It starts with an overview of the approach where a two-tier solution was adopted to address safety issues and then proceeds to detail each of the developed technologies.

Keywords: unmanned marine systems, unmanned surface vehicles

1. Introduction

During maritime search and rescue operations, the safety of the rescuers is a major issue and must be ensured in any circumstance. Therefore, these teams are often forced to adapt, or even to suspend their operations due to external factors and conditions, such as lack of visibility or atmospheric and/or maritime adverse conditions. On the other hand, it should be pointed out that rescue response time is a major factor for success in these operations, due to the reduced survival time of victims that fall overboard.

Robotic assets can therefore complement the role of search and rescue teams, as they can operate in dangerous scenarios and under adverse environmental conditions without putting human

lives in danger. There is nowadays a broad range of unmanned maritime systems (UMS) that can operate under different environmental conditions, transport a multitude of payload sensing systems and perform distinctive missions [1]. Concerning maritime robotic tools for search and rescue operations, two works are worth mentioning the emergency integrated lifesaving lanyard (EMILY) system [2] and the autonomous galileo-supported rescue vessel for persons overboard (AGAPAS) project [3]. EMILY (emergency integrated lifesaving lanyard) is a remotely operated autonomous vessel that aims to assist the life guards in crowded beaches, providing them a safe and fast response means. AGAPAS is a project orientated, specifically to person overboard situations, where an automatic system perceives that someone fell from the vessel and deploys an unmanned surface vehicle (USV) capable of fetching that.

While these systems are operated in an independent way, within the ICARUS project, particularly in the maritime scenario, multiple heterogeneous unmanned platforms (by air or surface) will co-operate, in order to detect and assist victims. This chapter addresses the adaptations of general purpose UMS and the development of novel assets, performed within the scope of the ICARUS project, to obtain an integrated system able to respond to search and rescue requirements in complex and challenging environments.

2. Overall concepts of operation and platforms

The assistance of UMS in search and rescue operations may include providing means for the floatation and thermal protection, preventing from fatigue, drowning or hypothermia, thereby increasing the survival rate. Furthermore, when the conditions do not permit the manned search and rescue operations, a fast and effective operation within the disaster scenario by the robotic assets makes it possible for the rescuers to evaluate and remotely assist the victims before resuming action as soon as the safety conditions are ensured.

Within the scope of the ICARUS project, a complementary approach for the use of UMS was followed. It consisted in having two classes of UMS, large and fast systems, able to arrive to the disaster area in a short time, and small and slower systems, able to get close to survivors on the water providing them floatation and thermal protection without putting them in danger.

For the larger and fast systems, two different platforms were considered: U-Ranger and Roaz II. U-Ranger is a 7 m long UMS, weighting more than 1000 kg and able to reach top speed exceeding 40 kts. Roaz II is a 4 m long, weighting up to 400 kg and able to reach a maximum velocity of 10 kts. These were existing platforms operated by partners of the ICARUS consortium, respectively, Calzoni and INESC TEC. Within the scope of this project, both these platforms were adapted for search and rescue operations, by integrating on them adequate sensor suites, by endowing them with autonomous behaviours suited for these operations and by incorporating on them the ability to carry and deploy on site smaller platforms.

The smaller platforms are the unmanned capsules (UCAPs) and were completely developed during the project and consist of 1.5 m long UMS, weighting up to 40 kg. Each of these vessels can be remotely operated or execute autonomous missions and carries on its deck an uninflated life raft. Upon reaching survivors, an automatic inflation of the life raft is performed allowing the survivors to jump it on board.

3. U-Ranger USV

The U-Ranger (**Figure 1**) is a remotely controlled unmanned surface vehicle (USV) mainly tailored for harbour and ship protection, able to perform intelligence, surveillance and reconnaissance (IRS) operations and patrolling of pre-defined areas.

The U-Ranger can be equipped with different kinds of sensors like cameras and radar for surface area control, sonar sensors for underwater control and other sensors for environment control. **Table 1** lists the main technical characteristics of the system.

Within the scope of the ICARUS project, the U-Ranger USV was equipped with a sensor and autonomous behaviour payload from Centre for Maritime Research and Experimentation (CMRE) [4]. The autonomous behaviour payload is based on the mission oriented operating suite (MOOS) open-source open architecture. The MOOS is a C++ cross platform middleware for robotics research. Its advantages include open source, flexibility, capacity for system growth, functionality across all platforms, a large user community who contribute MOOS architecture modules for all to share, scalability through distributed computation, protocols exist and need not be developed and considerable use in autonomous systems internationally, at CMRE and elsewhere. MOOS interacts with hardware and operator GUI through MOOS interface drivers and MOOS processes (autonomous behaviour sets), each of which is an independent process linked to a central MOOS database by standard internal process connections. Processes post data to the central MOOS database for access by any other process. Processes

Figure 1. Calzoni U-Ranger USV (source: ICARUS).

Boat type	RHIB type, full aluminium (included tubes)
Length	7 m
Beam	2.5 m
Weight	1400–1800 kg (depending on payloads)
Motor power	260 CV
Speed	>40 kts
Autonomy	>8 h
Safe operating conditions	Sea: SS3 Douglas
	Wind: 6 Beaufort
Control range (VHF)	Up to 15 nm
Wide band range	>5 nm

Table 1. Calzoni U-Ranger technical specification.

subscribe to the data they require, drawing it from the MOOS database on notification of updates. Communications between the sensor/behaviour payload and the control station on-shore rely on a worldwide interoperability for microwave access (WiMAX) link, while the very high frequency (VHF) link bypasses the sensor/behaviour payload allowing direct full manual control of the USV.

The sensor suite includes the following sensors:

- RADAR: Obstacle detection

- Laser scanner: Obstacle detection

- Weather station: *In Situ* weather data

- Daylight camera: Survivors detection

- Thermal camera: Survivors detection

The thermal and daylight cameras allow night and day operations. Their fields of view and resolutions are such that it is possible to detect a person in the water at 200 m. While the day-light camera is quite sensitive to lightning conditions and in particular to the reflections of sun light on the water surface, the thermal camera can provide useful data almost independently of the environmental conditions. An example of an image provided by this camera can be observed in **Figure 2**. Furthermore, the cameras are mounted on a gyro-stabilized platform that also allows their pan and tilt command.

The radar installed on the U-Ranger operates on the X-Band (9.3–9.4 GHz) and can be config-ured with different range settings from 50 m to 24 nautical miles. It has a rotation rate of 24 RPM and it is interfaced to the computational system using public domain C++/Java plugins/libraries (openCPN BR24 plugin and openbr24 Java). This radar is able to reliably detect obstacles at ranges greater than 50–100 m. The major drawback is the difficulty in detecting fast objects or making detections during sharp turns of the U-Ranger.
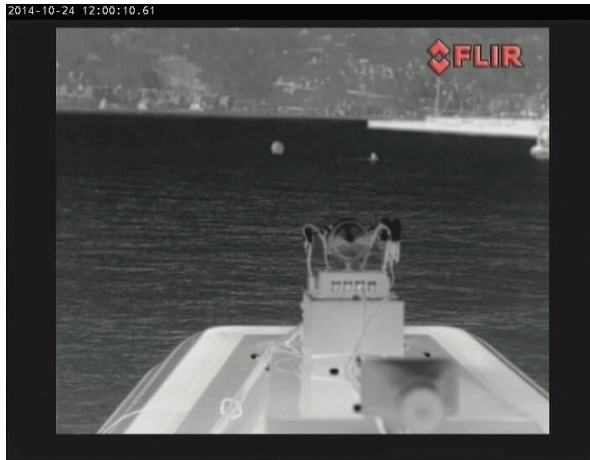
Figure 2. Example image from the thermal camera on the U-Ranger (source: ICARUS).

The laser scanner has four vertically stacked beams with a spacing of 0.8°, which are steered within a 110° angle and with 0.125° horizontal resolution. The maximum scanning frequency is 50 Hz and the obstacle detection range is greater than 100 m. This scanner is mounted near the U-Ranger bow on a gyro-stabilized platform (**Figure 3**).

The behaviour set implemented in the U-Ranger MOOS system contains the following elements:

- Constant speed
  - Classic part of the MOOS-IvP and always active.
- Station keeping
  - U-Ranger vehicle is not able to stop in an autonomous mode.
  - Vehicle with a single thruster and rudder.
  - Last waypoint is always considered as a station keeping point.
- Waypoint behaviour
  - Selection of waypoints is the C2I operator responsibility.
  - Any waypoint can be selected as the U-CAP deployment point except the last one..
- Operational region
  - Pre-defined operational area polygon.
- Obstacle avoidance

The available behaviours are combined in real time by a function optimizer in order to determine the direction the USV should take.
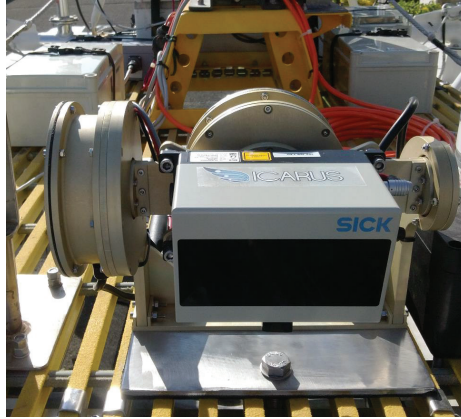
Figure 3. Laser scanner mounting platform and electronics (source: ICARUS).

4. Roaz II USV

Roaz II is an USV that can operate in full autonomous mode or remotely operated from a base station. It can be configured to carry different sets of sensors and to perform several kinds of missions, including environmental monitoring, harbour protection or bathymetric data gathering. Its main characteristics are described in **Table 2**.

Roaz II is operated from a mission control station composed by a ruggedized computer and a set of auxiliary devices including antennas. It is capable of executing autonomous missions defined by a list of waypoints differential GPS system and an inertial measurement unit. Telemetry as well as payload data are transmitted in real time to the mission control station.

Boat type	Autonomous marine surface vehicle/Roaz II
Length	4.2 m
Beam	2.0 m
Weight	200–400 kg (depending on payload and batteries)
Propulsion	Two independent electric thrusters
Speed	10 kts
Autonomy	>10 h (depending on operating speed)
Payload capacity	Weight: 200 kg (depending of installed batteries)
	Power: 1200 W
Control range (wide band)	Up to 1 nm

Table 2. Roaz II technical specification.

For navigation purposes, the vehicle uses a precision L1/L2 GPS receiver (Septentrio PolaRx2) and an inertial motion unit (Microstrain 3DM-GX1) providing attitude information. Propulsion is achieved through the use of two 2 kW electric thrusters with the vehicle reaching a maximum speed of 10 kts. A set of LiFePo batteries provides up to 8 h of autonomy. Communications with the control station are provided by a WiFi link, allowing remote control, autonomous mission supervision and also transmission of telemetry and payload data to shore.

Although this USV does not fulfil all the requirements for search and rescue (SAR) operations as defined in the ICARUS project (mainly the ones related to maximum speed and range) and its adaptation would require a great effort, its characteristics make it a valuable asset in many experiments as well as in the final demonstration scenario [5].

For that purpose, a thermal camera, a visible camera as well as a radar similar to the one integrated in the U-Ranger were also integrated in Roaz II (**Figure 4**).

Figure 4. Roaz II USV (source: ICARUS).

5. Unmanned capsule

The UCAP (**Figure 5**) is a single hull vessel, with a lower rear deck to accommodate the uninflated life raft as well as the corresponding compressed gas bottle. The hull was fabricated in fibreglass, using as custom-made mould [6].

The UCAP dimension is 1.45 m (length) × 0.52 m (width) × 0.42 m (maximum height). It weighs 22 kg and has a payload capacity exceeding 15 kg.

A jet drive unit assures the propulsion of the UCAP. This jet drive unit is attached to a brushless motor and is capable of delivering a maximum force of 80 N with a power consumption of 800 W. This maximum thrust assures a top speed greater than 5 kts.

On-board energy is provided by two packs of ZIPPY Flightmax 5000 mAh 6S1P LiPo batteries. This solution assures about 220 Wh of total on-board energy. Taking into account the efficiency of the propulsion system, the continuous operation at 1.5 m/s (3 kts) for 20 min (resulting in range of 2 km) should require about 100 Wh of energy, leaving 120 Wh for electronics and communications, which consume about 10 Wh. The battery pack is enclosed in a watertight box that is located in the bow compartment. This compartment also houses another watertight box with the on-board computer, navigation sensors and communications equipment. The bow compartment is also watertight assuring a double protection for electronics and batteries.

Navigation sensors include a PNI Trax AHRS and an Ublox Neo 6P GPS. Trax AHRS is a low-power and low-cost attitude and heading reference system with a static heading accuracy of 0.2° and an overall accuracy better than 2° in the presence of magnetic distortions. NEO 6P is a low cost GPS receiver that operates at 10 Hz, outputs raw data, being supported by RTKLIB, an open source library that implements differential and real time kinematic corrections using small size and inexpensive receivers.

Communications with a control station are assured by a long-range Wi-Fi link that establishes a wide band link over distances above 1 km (depending on height of the shore station antenna over the water surface and on the wave conditions).

Figure 5. Unmanned capsule (source: ICARUS).

A video camera is also installed on the UCAP. A video stream is fed to the control station for possible assessment of victim conditions when the UCAP is close to them, as show in **Figure 6**.

Besides the described items, on-board electronics also includes a load balancing and protection system for the batteries, the motor controllers or the water jet motor and direction servo as well as triggering systems for the inflation of the raft. The interconnections between all on-board systems are depicted in **Figure 7**.

The on-board software is composed of several modules that communicate with each other using a message passing mechanism as shown in **Figure 8**.

These modules follow a hierarchical architecture similar to the one used in the other INESC TEC robotic systems. At the lowest level, the modules that interact directly with the sensors and actuation devices constitute a hardware abstraction layer. On top of these, two major modules are responsible for the navigation (real time estimation of the UCAP position, velocity and attitude) and for the control (execution of manoeuvres and other high level behaviours).

The navigation module processes data from the GPS and inertial measurement unit (IMU) systems. The GPS system provides the information about location and velocity. The IMU has incorporated magnetometers, accelerometers and gyroscopes, providing information about yaw, pitch and roll states, acceleration and rotational movements decomposed. A data fusion algorithm is used to estimate the position of the capsule whenever the GPS receiver loses track, possibly due to excessive roll or pitch caused by stronger waves. At the same time, the inertial data are used to obtain updated information on the external disturbances, allowing a better characterization of the navigation environment.

The UCAP carries a lightweight life raft as the one presented in **Figure 9**. This life raft weights 8 kg (raft + full inflation bottle) and the overall volume before inflation is 13 dm^3.

Figure 6. Image taken from the unmanned capsule on-board camera (source: ICARUS).
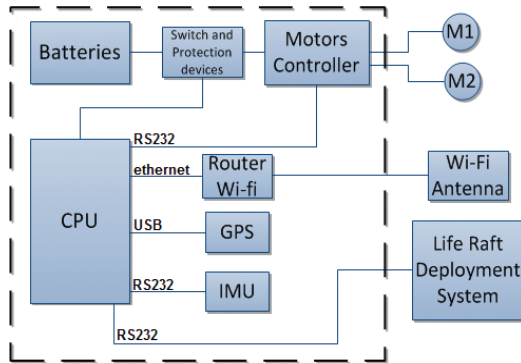
Figure 7. System architecture—electronics (source: ICARUS).
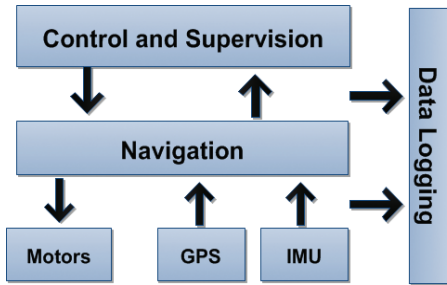
Figure 8. System architecture—software (source: ICARUS).

Figure 9. Inflated life raft (source: ICARUS).

A mobile computer running software for real-time monitoring and control of unmanned capsules composes the operation console. A graphical interface (**Figure 10**) provides the operator with the most relevant data concerning the state of a UCAP and allows him to control its operation. A joystick can be optionally connected to the computer to simplify the interaction with the UCAP.

This console allows the operator to switch between different UCAP operating modes:

1. Idle mode: where the UCAP actuation is shut down, causing it to drift according to external disturbances (winds and currents).

2. Anchor mode: that allows performing station keeping, where the UCAP will loiter compensating drifts caused by winds, currents or other influences.

3. Waypoint navigation mode: which consists of autonomous operation of the UCAP following a sequence of waypoints defined by the operator or imported from a previously defined file.

4. Remote control mode: where the operator remotely controls the UCAP.

5. External mode: which consists of granting the control of the UCAP to an external entity. This mode is similar to the idle mode except that another entity (for example the ICARUS C2I) can take control of the vehicle.
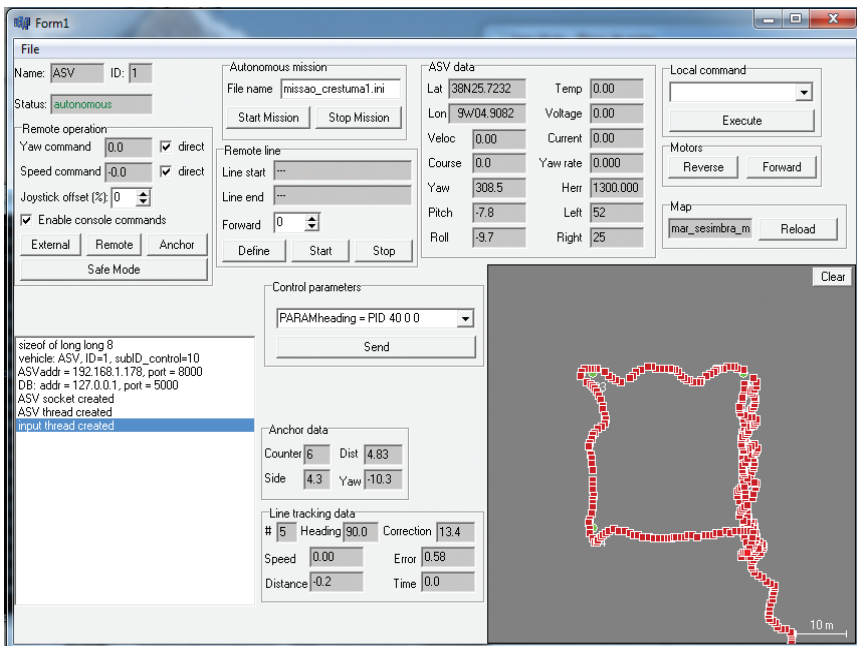
Figure 10. UCAP operation console (source: ICARUS).

Specific information about the UCAP that is displayed in the graphical interface includes

- Velocity

- Position

- Attitude (heading, pitch and yaw)

- Actuation (thrust and direction commands)

- Battery level

- Real-time power consumption and estimated endurance

When the UCAP is in autonomous operation, further information concerning the status of such operation (distance to next waypoint, estimated time no next waypoint completion or distance to anchor point in anchor mode) is also provided to the operator.

A heartbeat mechanism is implemented between the UCAPs and operator console to support emergency behaviour of the UCAP in case of communication link failure.

6. Unmanned capsule deployment system

The deployment system consists of a mechanical structure and a release system that could be easily modified or redesigned to suit several carrier USVs (**Figure 11**). The structure consists of a ramp that allows the gravity to be the main force imposing forward motion to the UCAP during launch. It is made of anodized aluminium bars to keep the overall weight low and resist to corrosion due to salt water. When placed on the launching ramp, the UCAP sits on rubber rollers that allow the movement in the forward direction while constraining it in the transversal direction [7].

The release system is composed by an electric latching device and an electronic system required for its command. This system is housed in a watertight box and is composed by a microcontroller, a power amplifier and a battery. The system is connected to the carrier USV communications infrastructure so that it can receive a remote command to release the UCAP. This command can be issued by a command line or a graphical interface running on any Linux-based device.

The deployment system can be easily integrated on a carrier USV (**Figure 12**), requiring only the following operations:

- Mechanical integration of the ramp

- Connection of the electronics box to the USV communications network

- Configuration of the microcontroller to use the carrier USV communications network

Mounting the UCAP on the deployment system is a simple operation that can be performed, whereas the carrier USV is moored next to pier (**Figure 13**). The UCAP can be directly mounted on the rollers on ramp with the help of a rubber boat.
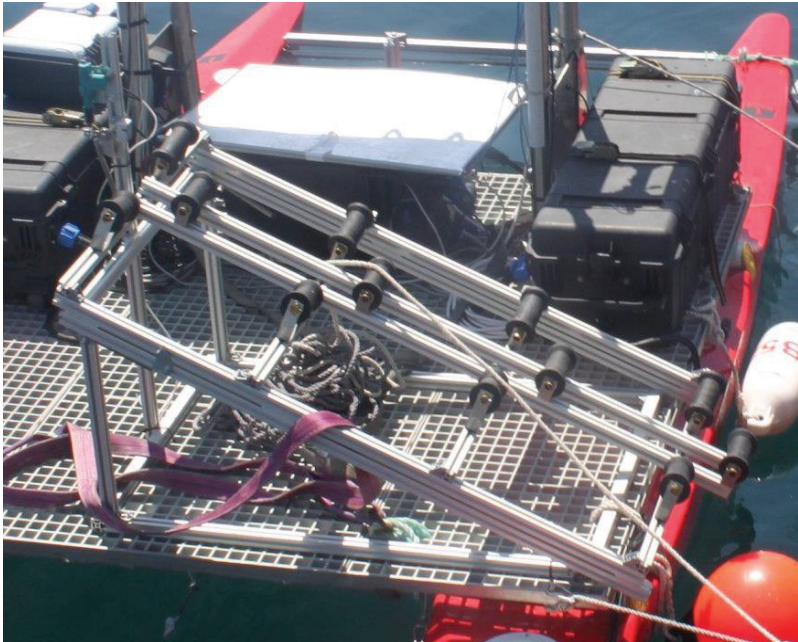
Figure 11. UCAP deployment system installed on Roaz II USV (source: ICARUS).
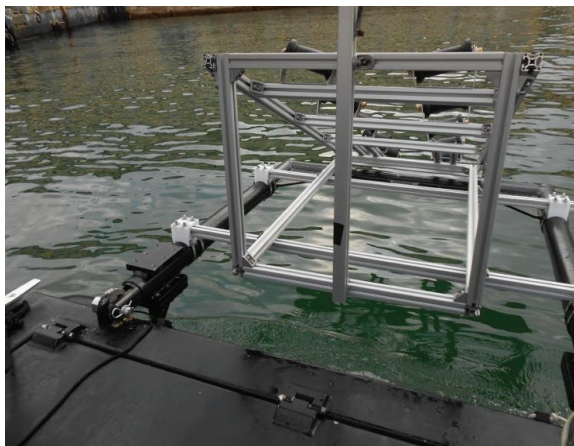
Figure 12. Deployment system installed on the U-RANGER USV (source: ICARUS).

Figure 13. Two UCAPs mounted on the deployment system installed on the U-RANGER (source: ICARUS).

Upon mounting the UCAP on the ramp, it must be secured to the release mechanism. For that purpose, a rope with a metal attached to it is fastened to the stern of the UCAP; and when the UCAP is in place, that pin is attached to the latching device.

Afterwards, for releasing the UCAP, a simple command needs to be sent to microcontroller in the electronic box of the deployment system (**Figure 14**).

Figure 14. UCAP being launched from the U-RANGER (source: ICARUS).

7. Conclusions

This chapter addresses the work performed within the scope of the ICARUS project on the development of complementary unmanned maritime systems technologies for search and

rescue. This work aimed at delivering a set of tools that can act not only as a part of the ICARUS toolset, but can also be used independently. For one side, such developments consisted in endowing medium and large scale unmanned surface vehicles with augmented perception and autonomic capabilities so that they could perform search and rescue operations in complex environments with the presence of other vessels and victims on the water, reporting back to the control stations situational awareness information. On the other hand, the concept of unmanned capsule, which is a small-size platform able to carry a life raft and inflate it close to victims, was prototyped, and integration as payload of larger unmanned platforms was also addressed. These developments and their extensive validation in several field trials and demonstrations carried out along the project are therefore a relevant contribution for the real-world deployment of robotics platforms in search and rescue operations, complementing the operation of traditional search and rescue teams.

Acknowledgements

Author details

Aníbal Matos[1,2]*, Eduardo Silva[1,3], José Almeida[1,3], Alfredo Martins[1,3], Hugo Ferreira[1], Bruno Ferreira[1], José Alves[1,2], André Dias[1,3], Stefano Fioravanti[4], Daniele Bertin[5] and Victor Lobo[6]

*Address all correspondence to: anibal@fe.up.pt

1 IINESC TEC – Instituto de Engenharia de Sistemas e Computadores, Tecnologia e Ciência, Rua Dr. Roberto Frias, Porto, Portugal

2 Faculdade de Engenharia da Universidade do Porto, Rua Dr. Roberto Frias, Porto, Portugal

3 Instituto de Superior de Engenharia do Porto, Rua Dr. António Bernardino de Almeida, Porto, Portugal

4 NATO Science and Technology Organisation – Centre for Maritime Research and Experimentation, Viale San Bartolomeo, La Spezia (SP), Italy

5 Calzoni S.r.l., Via A. De Gasperi, Calderara di Reno (BO), Italy

6 Escola Naval, Rua Base Naval de Lisboa, Almada, Portugal

References

[1] Caccia M. Autonomous surface craft: Prototypes and basic research issues. In: Proceedings of the 14th Mediterranean Conference on Control and Automation; 28-30 June 2006; Ancona, Italy: IEEE; 2006

[2] EMILY webpage [Internet]. 2017. Available from: http://emilyrobot.com. [Accessed: 31 March 2017]

[3] Clauss GF, et al. AGaPaS – Autonomous Galileo-supported rescue vessel for persons overboard. In: Proceedings of 28th International Conference on Ocean, Offshore and Arctic Engineering, OMAE 2009; 31 May–5 June 2009; Honolulu, HI, USA: ASME; 2009

[4] Fioravanti S, Grati A, Stipanov M. ICARUS – USV autonomous behaviour in Search and Rescue operations. In: Proceedings of RISE 2015 – 8th IARP Workshop on Robotics for Risky Environments; January 2015; Almada, Portugal; IARP; 2015

[5] Martins A, Dias A, Almeida J, Ferreira H, Almeida C, Amaral G, Machado D, Sousa J, Pereira P, Matos A, Lobo V, Silva E. Field experiments for marine casualty detection with autonomous surface vehicles. In: Proceedings of MTS/IEEE Oceans 2013 San Diego Conference; 23-27 September 2013; San Diego, USA: IEEE; 2013

[6] Ferreira B, Matos A, Alves J. Water-jet propelled autonomous surface vehicle UCAP: System description and control. In: Proceedings of MTS/IEEE Oceans 2016 Shanghai Conference; 10-13 April 2016; Shanghai, China: IEEE; 2016

[7] Matos A, Silva E, Cruz N, Alves J, Almeida D, Pinto M, Martins A, Almeida J, Machado D. Development of an Unmanned Capsule for Large-Scale Maritime Search and Rescue 2013. In: Proceedings of MTS/IEEE Oceans 2013 San Diego Conference; 23-27 September 2013; San Diego, USA: IEEE; 2013

Interoperability in a Heterogeneous Team of Search and Rescue Robots

Daniel Serrano López, German Moreno,
Jose Cordero, Jose Sanchez, Shashank Govindaraj,
Mario Monteiro Marques, Victor Lobo,
Stefano Fioravanti, Alberto Grati, Konrad Rudin,
Massimo Tosa, Anibal Matos, Andre Dias,
Alfredo Martins, Janusz Bedkowski, Haris Balta and
Geert De Cubber

Additional information is available at the end of the chapter

Abstract

Search and rescue missions are complex operations. A disaster scenario is generally unstructured, time-varying and unpredictable. This poses several challenges for the successful deployment of unmanned technology. The variety of operational scenarios and tasks lead to the need for multiple robots of different types, domains and sizes. A priori planning of the optimal set of assets to be deployed and the definition of their mission objectives are generally not feasible as information only becomes available during mission. The ICARUS project responds to this challenge by developing a heterogeneous team composed by different and complementary robots, dynamically cooperating as an interoperable team. This chapter describes our approach to multi-robot interoperability, understood as the ability of multiple robots to operate together, in synergy, enabling multiple teams to share data, intelligence and resources, which is the ultimate objective of ICARUS project. It also includes the analysis of the relevant standardization initiatives in multi-robot multi-domain systems, our implementation of an interoperability framework and several examples of multi-robot cooperation of the ICARUS robots in realistic search and rescue missions.

Keywords: interoperability, multi-robot collaboration

1. Introduction

There are nowadays many different types of unmanned systems being used in different domains and, certainly, this number will increase significantly in the upcoming years. In general, large scale systems aiming at solving all problems with one single type of platform have proven to be expensive and not flexible enough. Heterogeneous teams, composed by unmanned air, ground, surface, and underwater systems (UxS), of different types and sizes, offer the possibility to exploit the best features of each kind and combine them to obtain compound capabilities, which have demonstrated to be more cost-efficient and adaptable to new scenarios. Recent research efforts have focused on developing the autonomy of the team by increasing the interactions between these systems, making them aware of each other, executing tasks that require cooperation, and finally implementing flock or swarm coordinated behaviours.

The ICARUS project involves a team of assistive unmanned air, ground and sea vehicles for search and rescue operations. In order to effectively support the on-site person responsible for the operations, these systems must be able to collaborate as a seamlessly integrated team, coordinated from the ICARUS Robot Command and Control station (RC2) in the field.

A heterogeneous fleet is the one composed by elements of different kinds such as the ICARUS team, including up to ten different vehicles (long-endurance fixed-wing, outdoors multi-rotor, indoors multi-rotor, large UGV, small UGV, Teodor UGV, U-ranger USV, ROAZ USV, MARES AUV and several rescue capsules). Each robot has been developed by a different provider or partner, using its own design, framework and middleware. Thus, a strong effort had to be devoted to their integration as a team and this is the work described in this Chapter. Although many standards have been proposed by the community, most of the field robotic systems have their own command and reporting protocols, and consequently require their own ground control stations. This profusion of protocols makes the cooperation between systems difficult. The lack of unified standards poses an unnecessary burden on the operation and maintenance of multi-vehicle systems. The work described in this chapter aims at contributing to the harmonization of the multiple standardization initiatives for the coordination of heterogeneous teams.

The ultimate objective of the ICARUS project is to achieve robot interoperability, which can be understood as the ability of robots to operate in synergy to the execution of assigned missions. Interoperability enables diverse teams to work together, sharing data, intelligence and resources.

2. Approach to interoperability

Interoperability is the key that acts as the glue among the different units within the team, enabling efficient multi-robot cooperation. Seamless and non-ambiguous interaction between different robots of any provider and domain demands a common, well-defined interface.

ICARUS proposes the adaptation of all the vehicles to a single standard external interface as a method to ensure interoperability. Each robot development team is free to use their own tools inside their systems as long as the interaction with the rest of the team follows a set of definitions and rules referred to as the interoperability standard. This follows the façade pattern [1] very frequently used in software engineering. It essentially hides the complexities of the implementation and provides the outer components with a simpler interface. It is typically deployed as software library implementing a wrapper or adapter template. On one side, this library implements the interoperability standard interface and, on the other side, it provides a set of classes and functions (an API) for its integration with the specific middleware or software provided by each platform.

This approach may initially seem to reduce the level of integration among the agents if we compare it against natively sharing an internal protocol in all systems, but it promotes the maximum decoupling between the custom implementations, with its particularities, and the definition of the common interface. In the long term, this has shown to improve the seamless integration of the maximum number of systems and domains at a lower cost. The integration of new platforms into the team has literally been done in a matter of few hours during the project, provided that on-board hardware resources and communications are made available by the robot provider.

Therefore, the ultimate goal of the work on the heterogeneous team is to consolidate a common command, control and payload interface to be agreed and adopted by all robotics platforms and control ground stations (CGS) involved in an ICARUS operation. This approach provides a common framework for the development of collaborative unmanned assets, minimizing the integration time and costs by avoiding ad-hoc implementations.

There are other advantages in using interoperability standards. The use of a widely accepted interface helps to easily integrate new technologies with minor modifications to the existing systems. This facilitates the insertion of new technology for their operational use in the field, as end-users rely on proven technology and the preliminary validation will focus only on de-risking the new developments. Another advantage of the use of standards is that it will facilitate the backwards and forwards compatibility between existing and future vehicles and CGS provided by different providers. This can benefit companies to maximize the revenue from a specific product.

Our strategy in terms of interoperability is to build upon existing body of work in the field, avoiding duplicating and re-inventing proven technology. During the initial steps of the work, the most relevant multi-domain interoperability protocols for unmanned systems were identified and evaluated against the ICARUS end-user requirements and foreseen scenarios. During this phase, several collaborations with other European [2] and NATO [3] initiatives, together with the organization of workshops involving end-users and stakeholders, were extremely relevant to gather good quality information on the state of the art in the field.

2.1. Ontology definition

One of the challenges in multi-robot multi-domain interface standardization is to be able to embrace all type of systems, independently of their domain, particularities (i.e. size, operational

modes, etc.) or constraints (i.e. computational resources, communication bandwidth, etc.). Therefore, in order to methodically evaluate the existing initiatives, an analysis of the ICARUS robot specific interfaces control (ICD) and functional specifications (FSD) was performed to generate what we referred to as the project interoperability needs. Any information that was domain- or platform- specific was removed from the analysis to ensure the level of abstraction required to ensure standardization. Likewise, the needs were further developed through an analysis of other potential vehicles that could be integrated into the system in the future.

This set of needs has been formalized as an ontology. An ontology is 'an explicit, formal specification of a shared conceptualization' [4]. We use it to describe the set of concepts required to coordinate a multi-robot search and rescue operation. This includes concepts, at different levels from robots to systems, capabilities and sensors, and their relationships and assumptions. There have been previous and parallel efforts in this field. Namely, the IEEE Robotics and Automation Society (IEEE-RAS) created a working group named Ontologies for Robotics and Automation that aims at the definition of a core ontology for robotics and automation [5]. The work performed in ICARUS has a strict focus on heterogeneous multi-robot operations in search and rescue, and as such, it proposes an application-specific ontology, addressing tasks and platforms involved in search and rescue missions.

This analysis resulted in a description of the set of multi-domain concepts and relationships or messages commonly found in unmanned systems. **Table 1** summarizes the key categories and provides some example of interactions between systems.

The complete ontology was used in a gap-analysis for the evaluation of the existing standards as described in Section 3.

2.2. Interoperability levels

A key concept that enables interoperability among the largest number possible of unmanned systems is the levels of interoperability (LoI). This concept is introduced by STANAG 4586

Category	Description and examples
Transport	Inter-process communication such as send, receive, broadcast, etc.
Commands	Generic accessories such as set, get, etc. for any standard concept
Management	Heartbeat, system status, clock synchronization, alarms, etc.
Telemetry	Pose and velocity reports in appropriate system coordinates, etc.
Telecontrol	Teleoperation, waypoint and mission management, etc.
Perception	Imagery, ranging, audio, etc.
Manipulation	Joint and end-effector control of robotics arms
Mapping	Maps, digital elevation models, point clouds
S & Rintelligence	Sectors, disaster alerts, humanitarian information

Table 1. Examples of the ICARUS ontology concepts organized by categories.

[6] and has been adopted in ICARUS and adapted for the purpose of our project. LoI defines the different degrees of compliance with the standard interface. It proposes a mechanism to account for a large variety of approaches and levels at which different systems can be integrated, accounting therefore for more integration strategies and combinations.

STANAG 4586 defines LoI as 'the platform, subsystem or sensor ability to be interoperable for basic types of functions related to unmanned systems'. These levels show different degrees of control that a user has over the vehicle, payload or both. However, these definitions have been adapted to our project as follows.

Therefore, the levels of interoperability in ICARUS are defined as shown in **Table 2**.

The levels of interoperability for each of the ICARUS systems are as shown in **Table 3**.

2.3. Adjustable automation

ICARUS is, by definition, a human-centred designed system. One of the most critical end-user requirements is to ensure that a member of the search and rescue team in the field always supervises the robot operations to ensure safety and effectiveness. ICARUS robotics asset can generally be remotely controlled. Most of these systems also provide on-board autonomy modules that allow the operator to plan a mission to be autonomously executed by the system. This should presumably help reducing the workload of the operator. However, in a realistic scenario, unexpected events are highly likely to occur and the intervention from the operator, such as manually overriding the mission execution, is often required. This increases the cognitive workload of the operator leading to stress and potential mistakes, which are even more critical in the context of multi-robot operations.

Adjustable automation (AA) is the ability of a robot to behave autonomously and dynamically change its level of independence, intelligence and controllability to adapt to different tasks and scenarios [7]. AA presents advantages when dealing with communication delays, human workload and safety [8]. Having systems that can dynamically reduce or increase the level of automation running on board provides a more flexible and reliable system.

In ICARUS, AA is achieved by supporting multiple levels of automation in the robots, e.g. fully autonomous, guided by the operator, and fully controlled by the operator. The C2I also

Level of interoperability	
LoI 1	Indirect receipt/transmission of telemetry, control and payload data: the UxV data are received from (or sent to) another source (another CGS, web-server, etc.)
LoI 2	Direct receipt/transmission of UxV telemetry and payload data, but without control authority over it
LoI 3	Direct control and monitoring over the UxV without launch and recovery. A dedicated control station keeps control for the safety critical operations of the platform (i.e. take-off and landing, deployment and recovery, etc) and hand it over to the CGS once ready for mission
LoI 4	Highest level of interoperability. The CGS has full control of the UxV

Table 2. ICARUS definition of the levels of interoperability.

Level of automation per robot

	LoI 1	LoI 2	LoI 3	LoI 4	Notes
Long-endurance fixed-wing			X		Take-off and landing procedures for the
Outdoors multi-rotor			X		UAVs are handled by the proprietary control stations. The system is handed
Indoors multi-rotor			X		over to the ICARUS C2I once in air
Large UGV				X	
Small UGV				X	
U-ranger USV	X				U-ranger is a highly equipped and extremely fast USV. Integration is done through its proprietary CGS for safety purposes
ROAZ USV			X		ROAZ USV is primarily operated from a proprietary CGS. When ICARUS mission starts, control is handed over to the ICARUS C2I
Rescue capsule				X	

Table 3. LoI for each of the robots in ICARUS.

supports adjustable automation by automatically changing its display and control functions based on the relevance of the information, the current situation the robots encounter, and user preferences.

The level of automation of a robot is related to the degree of intervention of the human operator and other robots in the decision process. However, the fact that a robot is autonomous does not imply that it has to make all its decisions by itself. Different levels of automation and classifications have been described in the literature [9]. Specifically, Lacroix et al. [10] defines five levels of automation according to the robot responsibilities towards a fleet of robots (tasks allocation, mission coordination, etc), which are mostly relevant for tightly coupled coordination. In ICARUS, the levels of automation are understood in terms of tasks execution and are reduced to essentially three modes, as shown in **Table 4**.

An ICARUS platform can seamlessly carry out a given task at different automation levels, depending on the robot operator choice, the mission plan priorities, workload and constraints of the mission and platform. As mentioned before, the concept of adjustable autonomy implies the ability to adapt and dynamically change between these levels of autonomy depending on situational changes. Some examples of adjustable autonomy within the context of ICARUS are:

- A UAV may provide fully-autonomous navigation in nominal conditions, but may fall back to semi-autonomous navigation in the presence of victims detected on the sensor stream.

- The RC2 operator may have initially designed the mission to manually operate the outdoor multi-rotor to inspect a building, but operation enters a highly complex area and he/she decides to enable semi-autonomous mode to ensure all corners are correctly surveyed.

Level of automation	
Level 1	Teleoperation. No automation on-board the robot. The robot is directly controlled by the operator
Level 2	Semi-autonomous. Execution capabilities. The robot is able to manage partially ordered sequences of elementary tasks, and to return execution status of the tasks. An operator is supervising the mission from the RC2
Level 3	Fully-autonomous. Deliberative capabilities. Complex task requests are managed (tasks planning and scheduling)

Table 4. ICARUS definition of the level of automation.

Most ICARUS platforms provide all three operation modes. However, there are specific constraints in some platforms due to their size or domain. Namely, the large UGV is usually remotely operated or waypoint guided. Such a large system should not be tasked with predefined missions. On the other hand, the U-ranger is such a fast maritime system that the operator should rely on the on-board autonomy, which is equipped with collision avoidance functionality. Obstacles at sea are difficult to see by the operator and therefore, this system is better commanded at full automation. **Table 5** illustrates the automation levels available in each of the ICARUS platforms.

2.4. Multi-robot cooperation

An effective heterogeneous team management requires the capability to do reasoning about the mission goals in order to provide a task-to-robot decomposition. This task allocation must take into account the current capabilities and constraints of each asset in the team. Different strategies for the cooperation are feasible and, therefore, different requirements may be placed on the interface in order to implement these strategies. A heterogeneous team usually contains a set of vehicles with diverse capabilities that can therefore play different roles in the mission. Concepts such as roles, responsibilities, modes of operations and tasks may be part of the standard interface that supports the fleet interoperability.

The platforms involved in ICARUS have been carefully selected to help each other. In other words, they play complementary roles. Several ICARUS platforms grouped together form a team. Each vehicle has been designed to provide a set of specific functionalities but they can address more complex missions by supporting each other.

Level of automation per robot							
	Long-endurance fixed-wing	Outhoor multi-rotor	Indoor multi-rotor	Large UGV	Small UGV	U-Ranger USV	ROAZ USV Rescue Capsule
Level 1							
Level 2							
Level 3							

Table 5. Level of automation for each of the robots in ICARUS.

Different strategies for the coordination are feasible. In the case of ICARUS, a strong end-users need is that any planning decision must be authorized by the on-site operations coordinator. Therefore, according to a traditional classification of multi-robot systems based on the coordination strategy [11], ICARUS follows a supervised, weakly-coordinated, centralized approach where the cooperation and interaction between robots is negotiated during mission planning. The planning, coordination, and therefore the ultimate responsibility fall on the ICARUS team operator and occur at the C2I. Therefore, this coordination approach relaxes to a certain extent the need to have multi-robot related concepts in the interoperable interface. The C2I encapsulates this functionality and can interact with each asset individually. However, a standard for coordinated multi-robot operations remains extremely relevant and was taken into account in the analysis described in the next section.

Some of key concepts to be unambiguously defined as the basis for efficient mission planning are goal, role and task. A mission goal refers to the overall objective that the fleet must accomplish, for instance, the assessment of a disaster area. The mission planner is responsible for coordinating the fleet and allocating specific roles to each robot. A role defines the robot's behaviour and its interactions with other members of the fleet or with humans. A task is the basic unit describing the actions requested from a robot. Typically, the role defines which tasks a robot should and should not execute. A robot is defined by the type of robot and its capabilities. For instance, the long-endurance is a fixed-wing aerial platform with surveillance, mapping, victim detection and communications relay capabilities. These characteristics define the set of tasks that it is able to perform, and therefore the roles it can take. A mission plan is therefore built upon the concepts of roles, tasks and responsibilities.

ICARUS planning flow mirrors the concept of operation of international search and rescue teams. **Table 6** illustrates how goals to roles and task decomposition occur on the field.

One of the core services required from all the platforms is the dynamic discovery of features. It allows robots to advertise their capabilities over the network, enabling dynamic planning and

System	Responsibilities
C2I	Sectorization + mission goals definition
	SAR team allocation to sectors
	Robot(s) allocation to teams
	Teams monitoring and control
RC2	Operations scheduling
	Roles allocation
	Robot task planning
	Robots monitoring and control
Robot	Task plan execution
	Progress and status report

Table 6. ICARUS goals to roles and tasks decomposition.

supervision from the C2I based on the current state of the team. A robot may take different roles during a mission depending on the responsibilities that the C2I allocates to it. The allocation of mission goals to predefined roles, the decomposition of these roles into tasks, and the configuration of these tasks for a specific robot model are the responsibility of the mission planner. Some predefined profiles are available to facilitate this task. Whereas roles influence the robots' behaviour, tasks influence the actions that robots perform. They are defined as a set of actions. Each task could be decomposed into subtasks. This subdivision could continue iteratively until a primitive task is reached. **Table 7** shows some examples of the roles defined in the ICARUS concept of operations.

Roles	Description	Modes
Scout	Provides a quick assessment of an unexplored area or route	Overview of an entire disaster zone (fixed-wing UAS). Traversability/best route exploration (UAS)
Surveyor	Scan in detail an area or building to support a thorough assessment and inspection (structural integrity, victims, hazards, etc.)	2D/3D geo-referenced map of the entire disaster zone as basis for sectorization (fixed-wing UAS). High-resolution 2D/3D geo-referenced map of a sector (fixed-wing UAS at higher altitude, rotor-craft at lower altitude) or a structure (rotor-craft). Building indoor inspections (small rotor-craft and small ground vehicle)
Observer	Steady target observation and assessment, including victims and structures	Steady hover over a target (rotor-craft), including harsh weather conditions. Victim medical assessment outdoors (rotor-craft and USV) and indoors (small rotor-craft and ground vehicles)
Searcher	Victims search	Outdoors human detection on IR (UAS and USV), indoors (small rotor-craft and UGV)
Rescuer	Support to victim rescue	Helps victim to escape from hazard areas (large ground vehicle) or support human rescuers
Deliverer	Safety kit delivery. Robot delivery	Delivery of a survival kit to a victim, aerial (rotor-craft) or terrestrial (UGV)
Cruiser	Travel to a destination	All platforms when transiting to a new location where another role is enabled. The larger platforms may also act as a carrier of tools, debris and smaller robotics assets

Table 7. Examples of ICARUS roles.

3. Analysis of existing standardization initiatives

ISO defines a standard as a set of 'requirements, specifications, guidelines or characteristics that can be used consistently to ensure that materials, products, processes and services are fit for their purpose' [12]. In the context of interoperability, a standard shall unambiguously define data types, message and rules to implement the protocol. The analysis of the existing standardization initiatives shows that there exist several predominant initiatives for interoperability of unmanned systems [3]. However, harmonization among them is not yet a fact.

In the context of this analysis, we divided the different initiatives in two different groups:

- Fully operational standards and

- Partially operational resources.

The first group focuses on systems interoperability, providing a common communication framework between different agents. They provide all the basic functionality required for a multiplatform system. The second group includes initiatives that are, either very popular on specific fields, or are designed specifically for some particular tasks or domain. Most of them show some relevant contributions but they do not provide interoperability for all the possible types of platforms, systems and range of application.

The lack of a single standard of reference for interoperability of unmanned systems makes any choice difficult since it will have an impact one way or another on legacy platforms. However, some alternatives may fit better for a given set of requirements. Harmonizing the existing standards, by combining them into one, or by proposing a brand new standard, would obviously solve most of the problems, but it would have serious implications both in industry and other programs that have adopted them as their standard [13]. This is clearly beyond the possibilities of the ICARUS project on itself.

Along the studies, two candidates stood out from the rest, STANAG 4586 [14] and others related, and the Joint Architecture for Unmanned Systems—JAUS [16]. They are both stable, widely used and complete. STANAG pays a strong attention to the intelligence, surveillance and reconnaissance (ISR) data, while JAUS is instead more devoted to command and control interfaces of the platforms, robot navigation and perception.

In this context, both were created to address specific requirements in different domains. STANAG related standards are predominantly military and, even though they have been promoted for civil applications, their requirements are heavily demanding in terms of compliance. STANAG 4586 is mostly focused on UAVs, even though some other types of unmanned systems have been developed to meet this standard. It is perhaps very relevant for the interoperability of military assets across the different NATO members, but it is hard to be adopted by civil or research platforms without a strong investment. For instance, certifying a small multirotor UAV for the STANAG 7085 Interoperable Data Links for Imaging Systems is costly and probably a barrier for small platforms providers. Furthermore, the geographical constraints (NATO only), the focus on the bigger systems and the absence of open available implementation make this option less convenient. Likewise, JAUS was originally designed for UGVs. It is fair to say that JAUS has done great efforts to extend the coverage to any type of platform, and it currently considers any unmanned system as a generic asset in order to become truly multi-domain. Its root is also military, but it was soon transferred to the Society of Automotive Engineers (SAE International) where it is currently hosted.

According to our analysis, JAUS is fairly aligned with the needs of small unmanned platforms in terms of the interoperability described in Section 2. Also, JAUS has been successfully demonstrated in recent years for collaborative UAV-USV cooperative missions [15]. A quite direct traceability between ICARUS needs and the JAUS service sets is easily

derived. It is already compatible with popular transport protocols (TCP, UDP, serial) independent of the communication link beneath it, which makes it more flexible. And it is already multi-environment (air, ground and maritime). There exist both commercial and open source implementations. Unfortunately, there is a fee to access the JAUS documentation which may prevent some providers from using it. Nevertheless, the cost is deemed reasonable.

There are many other initiatives with strong support in different communities. According to the principles and needs for standardization defined above, these are considered software frameworks and middleware rather than full standards. For instance, the robotic operating systems (ROS) is used nowadays in many multi-robot systems. However, open-source initiatives are open and flexible by definition which may not provide the expected reference specification for future developments. These initiatives definitely add a lot of value to the development of small-unmanned systems, but they do not formally satisfy the interoperability requirements like the standards mentioned previously. They should remain at the platform level and the platforms should comply with an external interoperability standard. It is the scope of the interoperability work to harmonize this heterogeneity into a single standardized protocol.

4. Interoperability standard

The ICARUS standard interface for interoperability of heterogeneous fleets is based on the Joint Architecture for Unmanned Systems (JAUS) [16]. JAUS is a service-oriented architecture (SOA) that specifies a list of services commonly found in robotics. ICARUS interface describes the subset of standard messages that will be used in the ICARUS scenario and specifies all the details required to comply with the ICARUS interface.

4.1. Service sets

The interoperability interface is a service-oriented architecture (SoA). The most common services for unmanned systems interoperability are already defined in JAUS as a set of advanced standards. They are grouped as 'Service Sets'. The following ones will be used in ICARUS:

- Core Service Set (SAE AS5710 [17]): essential services such as transport, events, discovery, etc.

- Mobility Service Set (SAE AS6009 [18]): mobile platforms services.

- Environment Sensing Service Set (SAE AS6060 [19]): platform-independent sensor capabilities.

- Manipulator Service Set (SAE AS6057 [20]): platform-independent capabilities common across all serial manipulator types.

The concepts defined in the ICARUS data model can be matched against specific services in this architecture. **Figure 1** shows the specific services used in a real ICARUS operation.
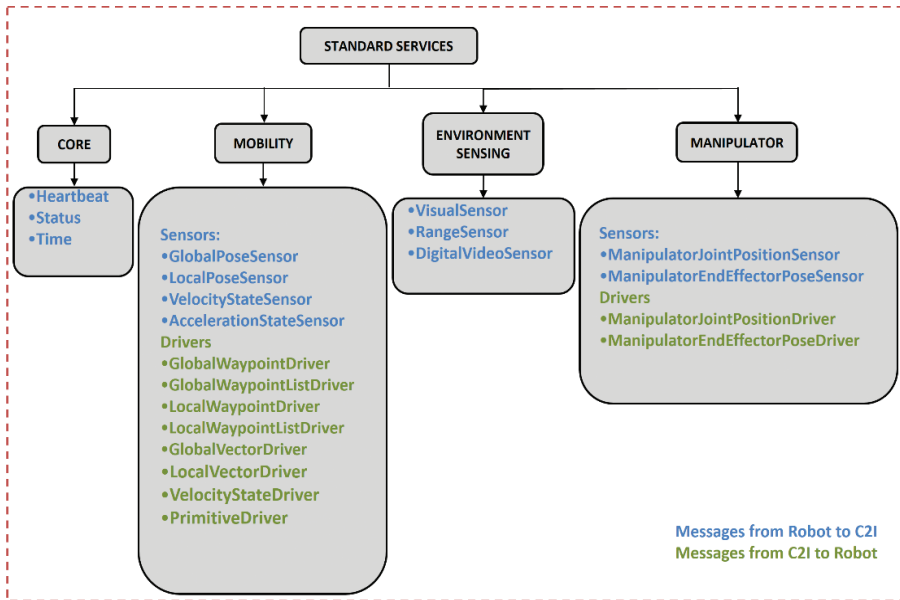
Figure 1. Relevant JAUS services (source: ICARUS).

However, as we progressed with all the integrations in the project, we discovered that some of the functionalities provided by some of the platforms were not supported by these standard services. We refer to this as the gap analysis. **Table 8** shows some of these gaps.

Given this, a new set of non-standard services has been defined to fill the gaps of the standard. This new non-standard service set is shown in the following **Figure 2**.

Gaps analysis				
	Outhoor quadrotors	Indoor quadrotors	Ground robots	Large sea vehicles
Survival kit deployment				
Rescue capsule deployment				
Platform-specific components enable/disable				
Platform extended status				
Manipulator tool selection				
Voice transmission				

Table 8. ICARUS interface gaps.
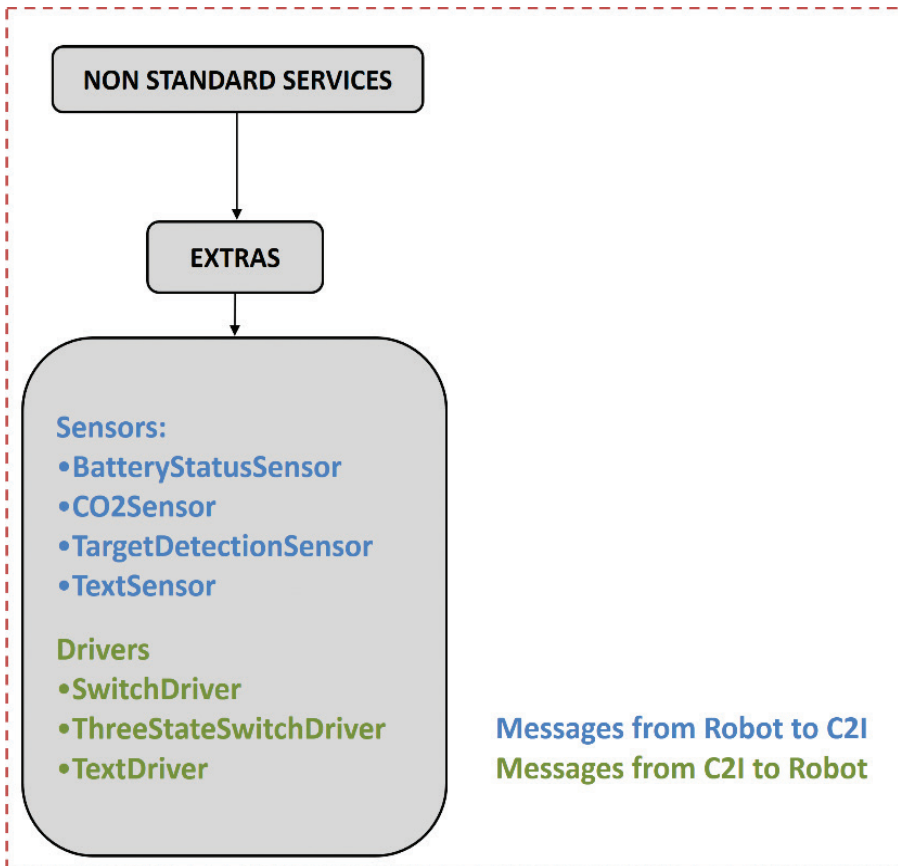
Figure 2. Additional non-standard JAUS services (source: ICARUS).

For each service, a strictly defined message-passing interface (vocabulary) and protocol (rules) for data exchange are available. There are generally three types of messages: query, report and command. Furthermore, the transport service (from the core service set) acts as an interface to the transport layer. Therefore, ICARUS interface is, in principle, independent from the physical transport layer. However, the current implementation for ICARUS is only available for the UDP protocol.

JAUS also defines a hierarchical and flexible topology built up of subsystems, nodes and components. For the implementation of JAUS within ICARUS, the following assumptions have been made:

- An ICARUS team is considered a system,

- Each platform is defined as a subsystem with a single node. Therefore, all components within the same platform will share the subsystem and node identifiers.

- As described later, a node may contain several components. But a component will implement only one service, plus the core services, which are always present. This restriction allows the C2I to dynamically discover each of the services available on each robot.

Therefore, an ICARUS system is depicted in **Figure 3**.
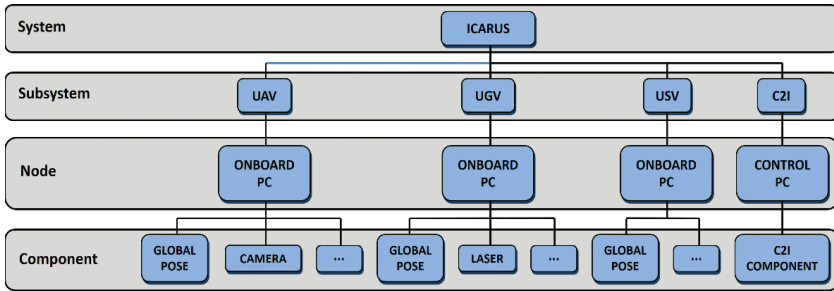
Figure 3. ICARUS JAUS topology (source: ICARUS).

5. An interoperable layer: the ICARUS library

All this functionality is provided to the ICARUS robotics partners as a software library referred to as the ICARUS interoperability layer. This module acts as a bridge between their internal and external development frameworks. This interoperability layer is also responsible for the integration of the ICARUS communication network and the command and control station on each individual platform.

A set of C++ classes has been designed to integrate the vehicles into the ICARUS network. The JAUS-specific functionality has been encapsulated within them. To comply with the ICARUS interface, a system may directly integrate this library (native integration). However, most robotics systems nowadays are based on either proprietary or open-source middleware (such as ROS). To accommodate these systems into an ICARUS compliant network, an alternative is to implement an adapter to the robot-specific middleware (translator). The diagram in **Figure 4** illustrates both cases, native integration (Robot C) and through an adapter (Robot A using ROS, and Robot B using MOOS).

The following sections describe the software classes encapsulated within the library and depicted in the previous diagram.

5.1. JAUS robot

JAUS robot encapsulates all the functionality required on-board the vehicle. It represents a subsystem in the JAUS topology (see **Figure 3**). In the ICARUS JAUS interface, a subsystem will
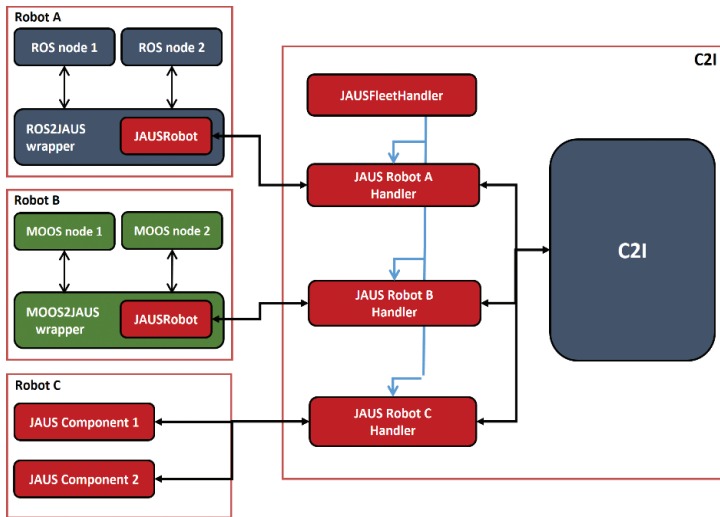
Figure 4. Robot adaptation strategy (source: ICARUS).

contain only one node. This approach will allow us to provide a component name to each service. Therefore, all components within the same robot share the subsystem and node identifiers.

There are two types of services available on a JAUS robot in addition to the core services: sensors and drivers.

Sensors provide access to information generated on the robot (i.e. global pose, image, etc.). There are essentially two types of C++ functions required to integrate this functionality:

- Add sensor: a JAUS component is added to robot subsystem. For example, if the instance of JAUSRobot is myRobot, the following statement adds a GlobalPoseSensor service to our robot:

    JAUSRobot myRobot;

    myRobot.AddGlobalPoseSensor ('OEMStar_GPS', AT_1_HZ);

- Set data: updates the data associated to a service. For the example above, the following lines will update the current GlobalPose of myRobot:

    JAUS::GlobalPose globalPose;

    myRobot.globalPoseSensor->SetGlobalPose(newGlobalPose);

Drivers, on the other hand, provide access to actuation capabilities provided by each robot (i.e. go to waypoint). There is an equivalent function required to integrate this functionality:

- Add driver: a JAUS component is added to the robot subsystem. For example, if the instance of JAUSRobot is myRobot the following line is adding a AddGlobalWaypointDriver service to receive waypoints request in global coordinate frame:

> *JAUSRobot myRobot;*
>
> *myRobot.AddGlobalWaypointDriver('global_waypoints', authority_code);*

The authority code parameter of these services is used for pre-emption and needs to be set lower to the one of the client accessing the driver. Otherwise, commands from the client are ignored.

Therefore, JAUSRobot creates a JAUS component for every new Sensor and Driver. This allows the JAUSFleetHandler class to discover and manage each of them independently.

The ICARUS JAUS interface is based on callbacks for message reception. One more function will register a local callback in order to receive any message coming from the JAUS network:

> *void localProcessMessage(const JAUS::Message* message){ }*
>
> *myRobot.RegisterJAUSMessageCallback(localProcessMessage);*

5.2. JAUS C2I

On the C2I side, two classes have been designed:

5.2.1. JAUS fleet handler

JAUS fleet handler encapsulates all the functionality related to the fleet management. It includes the functionality to discover subsystems and services on the JAUS network and retrieve their names and their current status. For example, if the instance of JAUSFleetHandler is myFleet, the following line allows discovering all subsystems on the JAUS network and retrieving their services names:

> *myFleet.DiscoverFleet();*

On the other hand, the following line also allows checking for system updates:

> *myFleet.RefreshFleet();*

In terms of JAUS, it represents a basic JAUS component implementing the discovery service to retrieve the subsystems available in the network and their services.

5.2.2. JAUS robot handler

JAUS robot handler is responsible for managing a single robot. After the discovery process, an instance of this class must be created and configured. This class will interface directly to the JAUS robot.

In terms of JAUS, it represents a basic JAUS component. For each sensor service available on the real robot, it creates an event of type every change. This is the JAUS mechanism to configure the sensor service on-board the robot to send data for every new set. Periodic events will also be available in the future.

On the other hand, for each driver service available on the real robot, it configures the access control service needed to send commands.

5.2.3. Management of the ICARUS communications

ICARUS tackles interoperability at different levels. This chapter focuses on the interoperability layer. This is a software-defined protocol (SDP) that can run over any compatible communication layer underneath. However, ICARUS also addresses the interoperability at the communications level, which is further detailed in Chapter 6 of this book.

A tight cooperation between the interoperability and communications layer allows for a smart management of the network. The ICARUS communications are a cognitive self-organizing multi-node network. This layer exposes an interface to configure the required data flows, its priorities and other details. Given the current set and number of robots, sensors and the priorities for the mission assigned, the communications layer can assure a quality of service for each data stream.

This is traditionally preconfigured manually for each mission. In ICARUS, a tight integration between the interoperability and communication layers has enabled the dynamic self-configuration of the team. A team management node runs within the C2I and exploits the discovery mechanism to retrieve all robots and their capabilities. This information is transferred to the communication layers that organized the data flows based on a-priori defined priorities. For instance, telemetry and telecontrol streams are given the highest priority since they are safety critical. For each robot, a camera is given the medium priority and all other sensors are lower priority. This a-priori priority allocation depends on the number of robots and their characteristics.

These configuration capabilities are also exposed to the C2I and therefore to the operator. The coordinator can at any time change the priority levels, enable new sensors, disable sensors that are not required, etc. The user is also informed with the current status of the network and he is therefore able to change the configuration if the network is overloaded.

Eight messages have been defined to exchange all the information needed between both interfaces (COMMS and JAUS):

- Register message.

- Activation of a stream.

- Deactivation of a stream.

- Deactivation warning.

- Network status notification.

- Robot's register notification. This is a notification sent from the COMMS interface to the JAUS interface. It is used to ask for the robot's register message presented before. It is needed when, for instance, the COMMS interface starts running later than the JAUS interface.

- Robot's unregister notification (COMMS to JAUS). It is sent when the COMMS interface loses a robot.

- Robot's unregister notification (JAUS to COMMS). It is sent when the JAUS interface loses a robot.

6. Individual platform adaptations

All ICARUS platforms have been adapted to the ICARUS interface. This automatically ensures the compatibility with the ICARUS C2I, enabling the multi-robot coordination and combination of data as later described in this chapter.

6.1. ROS to ICARUS robot node

All aerial platforms within the ICARUS project share a similar approach when it comes to software and hardware design. The hardware setup comprises a low-level board, responsible for the flight control of the vehicle (i.e. autopilot) and, optionally, a high-level board (i.e. on-board pc) responsible for the autonomous navigation and payload data. These autopilots communicate with the vehicle-specific ground station through the MAVLink protocol [21]. The on-board PCs, instead, run the robot operating system (ROS). A template has been developed to integrate ROS-based platforms. Therefore, all of them have been adapted using this template.

This template however should be configured to the specific characteristics of each platform, particularly in terms of sensors equipment. The proposed strategy is to implement a ROS-based wrapper to subscribe to ROS topics and interface the ICARUS protocol. This node is intended to run on-board the robot and provides a wrapper of ICARUS interface. The node is implemented within the robot.cpp file.

All the required services described in **Figure 1** are available for ROS-based systems through this template launch file. These services can be easily enabled by configuring a template ROS launch file. The XML excerpt below shows an example for the specific case of a global pose service:

```
<?xml version="1.0"?>

<launch>

   <node pkg="ros2jaus_node" type="robot" name="robot" output="screen">

      <!--ROBOT CONFIG -->

      <param name="subsystemName" type="string" value="ctae_robot"/>

      <param name="subsystemID" type="int" value="99"/>
```

```
    <param name="nodeID" type="int" value="1"/>

    <!-- GLOBAL POSE -->

    <param name="globalPoseEnable" type="bool" value="true"/>

    <paramname="globalPoseSensorName"type="string"value="global_pose"/>

    <param name="globalPosepUpdateRate" type="double" value="25"/>

    <param name="globalPoseTopicName" value="/EURECAT_robot/global_pose"/>

  </node>

</launch>
```

Therefore, a ROS-based robot just subscribes to a set of topics, with a predefined message type. An analysis of the existing ROS messages was performed to select the most appropriate interface definition. When an existing ROS message was deemed both valid and correct, this message was used. However, there were certain cases where the definition of the messages in ROS was either not existing, ambiguous or not valid. In these cases, a new message type was defined under the package icarus_msgs. The topic names and update rates can be configured from the ROS launch file:

- /global_pose (icarus_msgs/GlobalPoseWithCovarianceStamped)

- /local_pose (geometry_msgs/PoseWithCovarianceStamped)

- /velocity_state (geometry_msgs/TwistWithCovarianceStamped)

And publishes:

- /local_waypoint (icarus_msgs/LocalWaypointStamped)

- /global_waypoint (icarus_msgs/GlobalWaypointStamped)

6.2. Other robot adapters

The control systems of the ground platforms are implemented using FINROC, a middleware developed by the Robotics Research Lab at the University of Kaiserslautern. The rescue capsule runs OceanSys and exposes a data repository over which any software in the same network can subscribe and receive custom messages. ROAZ implements a similar system, but it is also ROS capable. The U-ranger on-board autonomy is developed in MOOS and implements a behavioural control strategy.

A template example was provided to each of the partners, together with the use case for ROS and some documentation. Each partner was able to quickly adapt its existing frameworks to the standard interoperable interface and become complying with all ICARUS team technology, such as the communication infrastructure and the C2I. **Tables 9** and **10** provide a summary of the ICARUS services provided by each system.

ICARUS services per robot

SERVICE	AROT	ASOLAR	U-RANGER	ROAZII	UCAP	FIREFLY	SUGV	LUGV
Global pose sensor	Global pose (20 Hz)	Global pose (20 Hz)	Global pose (20 Hz)	Global pose (20 Hz)	Global pose (20 Hz)	Global pose (20 Hz)	Global pose (20 Hz)	Global pose (20 Hz)
Velocity state sensor		Velocity state (20 Hz)						
First camera	Left camera (20 Hz)	Visual camera (20 Hz)		Visual camera (20 Hz)	Visual camera (20 Hz)	Left camera (20 Hz)	Front camera (20 Hz)	Front camera (20 Hz)
Second camera	Right camera (20 Hz)	Thermal camera (20 Hz)		Thermal camera (20 Hz)		Right camera (20 Hz)	Manipul camera (20 Hz)	Manipul camera (20 Hz)
Third Camera	Thermal camera (20 Hz)					Thermal camera (20 Hz)	Rear camera (20 Hz)	Gripper camera (20 Hz)
Fourth camera	Visual camera (20 Hz)							
Range sensor (Point clouds)			Laser	Radar		Point cloud	Point cloud	Point cloud
Global waypoint list driver	Global waypoints	Global waypoints	Global waypoints	Global waypoints	Global waypoints		Global waypoints	Global waypoints
Primitive driver	Cmd velocity		Cmd velocity		Cmd velocity	Cmd velocity	Cmd velocity	Cmd velocity
Manipulator End effector pose sensor							End effector pose	End effector pose
Manipulator joint sensor							Joint position	Joint position
Manipulator end effector driver							End effector pose control	End effector pose control
Manipulator joint position driver							Joint position control	Joint position control

Table 9. ICARUS services provided by each vehicle.

ICARUS custom services per robot

SERVICE	AROT	ASOLAR	U-RANGER	ROAZII	UCAP	FIREFLY	SUGV	LUGV
First switch driver	Delivery Kit				Deploy UCAP		Lights	Lights
Second switch driver					Inflate-raft		Manipulator	Manipulator
Third switch driver							Reset	Reset
Fourth switch driver							Audio	Engine
Fifth switch driver							Speech	Tool-Lock
First 3-state switch driver							Gripper control	Gripper control
Second 3-state switch driver								Tool selector
Text sensor							Text	
Text driver							Text Cmd	
CO2 sensor							CO2 sensor	
Robot extended status	Battery status	Battery status			Battery status		Battery status	Battery status
Video stream			Video stream					
Target detection	Victims	Victims		Victims			Victims	Victims

Table 10. ICARUS custom services provided by each vehicle.

7. Field validations: examples of multi-robot cooperation

The ICARUS approach to interoperability was initially verified with a set of in-lab integrations and simulated tests. Once the robotics platforms were finalized and ready for field tests, a series of field operations involving different combinations of pairs of air, ground and sea vehicles were organized during the integration trials carried out between July and September, 2014. The purpose was two-fold: (i) verification of the completeness, correctness and feasibility of the ICARUS interoperability interface; (ii) experimentation on the possibilities on multi-robot cooperative search and rescue missions. The results from one of these trials showed that the work on interoperability enabled large-scale cooperative mapping with multiple aerial and ground robots in urban search and rescue [22].

The final field validation was carried out together with the final integration and demonstration exercise of the ICARUS project described in Chapter 10. Three full-team validations were performed during the final project demonstrations:

- the maritime trials and demonstration in Alfeite, Lisbon (Portugal) in July 2015,

- the land trials and demonstration in Marche-en-Famenne (Belgium) in August 2015 and

- the participation in the euRathlon competition in September 2015 where the project received the Best Multi-Robot Coordination Award by the IEEE Robotics and Automation Society (RAS).

At this stage, all platforms had been integrated into the ICARUS system. After start-up, the current capabilities of the team could be dynamically discovered and the ICARUS C2I automatically configures itself, allowing an ICARUS team operator to plan a mission, assigning roles and tasks to each system. During the mission, all the information flows and the current network status are displayed. The operator can follow the progress of the mission, enable, disable or change the update rate of each of the ICARUS services. The operator can, at any time, request new missions, take manual control of the platforms that provide this service, resume previous missions, etc. All this functionality was exercised and demonstrated during the validations.

Together, these large-scale operational exercises completed the validation of the ICARUS interoperability standard interface. Therefore, ICARUS as a project has demonstrated multi-domain multi-robot heterogeneous interoperability in realistic search and rescue operations.

Some examples of the multi-robot collaboration experimented during ICARUS are described in the subsections below to illustrate the possibilities of multi-robot cooperation provided by an interoperable team.

7.1. Cooperative multi-stage aerial surveillance

In this multi-robot collaboration concept, the overall mission imposed by the human commanding officer is to provide a general assessment of a predefined sector. This is a typical scenario to be performed when relief agencies arrive on a crisis site. Assets to be deployed for this task are the fixed-wing UAV and the outdoor multi-rotor, both with clearly distinctive roles:

- The fixed wing aircraft acts as a surveyor system which covers the entire area quickly, flying at an altitude of around 100 m. Note that the altitude limitation is deliberative, as many countries impose a maximum flight altitude of 400 feet (133 m) for unmanned systems and it was our specific target to test the operational capabilities within realistic legislative bounds. Flying at this altitude, the aircraft quickly provides a general low-resolution assessment of the sector, such that areas of interest can be selected.

- The multi-rotor aircraft also acts as a surveyor system, but as it has a lower flight altitude of typically 40 m, it covers only a much smaller area. It is therefore used to provide a high-resolution assessment of an area of interest, as identified by the fixed-wing assessment.

- The multi-rotor aircraft also acts as observer system to provide high-resolution multi-view observation of points of interest (victims, buildings, etc.).

Operating multiple unmanned aerial systems in the same airspace is not easy from a safety perspective. In this case, vertical separation of the airspace was used for segregating the operations with the fixed wing and multi-rotor aircraft. The operators were also in constant contact with one another to synchronize the landing operations.

Figure 5 shows the different aerial systems in the air at the same time, whereas **Figure 6** shows the outcomes: the lower-resolution assessment by the fixed wing aircraft and the high-resolution assessment by the multi-rotor aircraft. As all data is geo-referenced, this information can be perfectly super-imposed on one another.

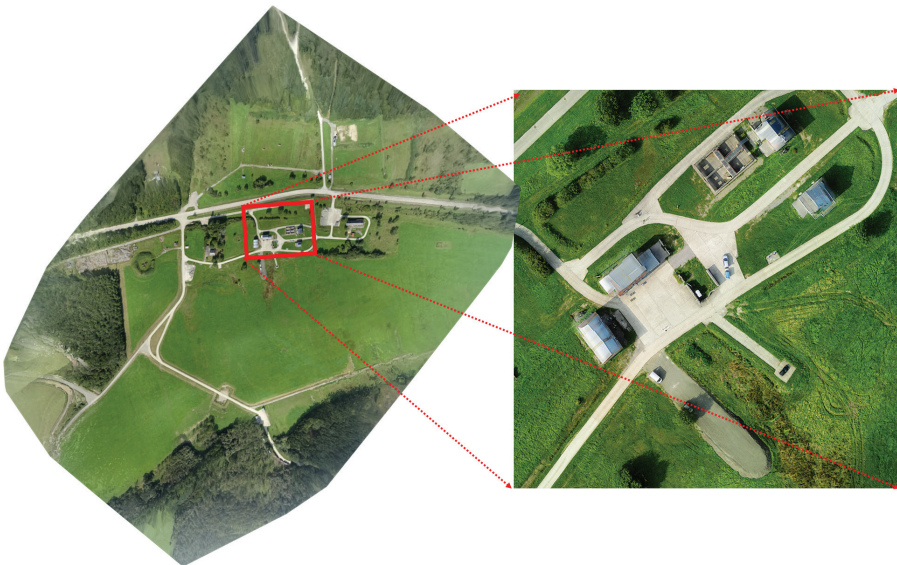**Figure 5.** Multiple ICARUS aircraft in the air at the same time (source: ICARUS).

Figure 6. Low-resolution fixed wing assessment and high-resolution multi-rotor assessment (source: ICARUS).

7.2. Aerial scouting for traversability analysis

A common problem for relief teams is that the route they have to take due to their designated destination cannot be reached using the 'normal' routes, as roads are blocked by debris or by floods. The mission resulting from this use case is to use a multi-robot team to ensure the traversability of a route and provide an early identification of threats. The assets deployed for this mission are the fixed-wing and outdoor multi-rotor aircraft and the relief team on the ground, including ICARUS unmanned ground vehicles. The aircraft scans the area to detect blocked and cleared routes to the destination point and sends updated navigation information to the ground team, such that the ground team can travel to the destination as quickly as possible. **Figure 7** shows the ICARUS multi-rotor aircraft flying ahead of the ground team, searching for obstacles on the way to the destination.

Figure 7. ICARUS outdoor rotorcraft ensuring optimal routing of ground team (source: ICARUS).

7.3. Victim search

Victim search is a primary mission in any rescue operation. Following the typical mission profile, a search area is defined and a multi-robot collaborative search is ordered by the human commanding officer in this predefined area. Multiple collaboration modalities are possible, depending on the search and rescue context:

- In outdoor search and rescue scenarios, the fixed-wing and outdoor multi-rotor aircraft are deployed. The fixed wing aircraft can quickly provide a scan of a large area, either clearing the area or indicating preliminary detections, which then need to be confirmed by the outdoor multi-rotor aircraft. The latter then confirms the location and status of the detected

victims and can count the number of victims. This operation can take place in an urban search and rescue context, where victims are to be sought in rubble fields after an earthquake (as shown in **Figure 8**), or in a maritime search and rescue context, where victims have to be found in the water (as shown in **Figure 9**).

- In indoor urban search and rescue scenarios, the indoor multi-rotor aircraft and the small-unmanned ground vehicle are deployed to collaborative search for surviving victims inside semi-demolished buildings, as shown in **Figure 10**.

- In maritime search and rescue scenarios, the survival times in the water are often very short. Therefore, in an attempt to limit the time-delay between the search phase and the rescue phase in the relief operation, the fixed-wing and outdoor multi-rotor aircraft are deployed, together with the ICARUS unmanned surface vehicles and the unmanned capsules. This enables the unmanned aircraft to immediately steer the maritime vehicles towards the victims detected and localized by the aircraft. The unmanned surface vehicles also have sensors (infrared cameras) enabling victim detection on board, but as these are relatively small platforms, their field of view in rough sea conditions with high waves is limited. Collaborative victim search between aerial and marine platforms is therefore not impossible, but the greatest benefit of a mutual deployment lies in the combination of the search and the rescue aspect, as illustrated by **Figure 11**, which shows a victim in the water being tracked and localized by the outdoor multi-rotor aircraft, thereby guiding the unmanned surface vehicle to a position in the neighbourhood of the victim, such that an unmanned capsule can be deployed, which can inflate a life raft close to the victim in order to save the person.

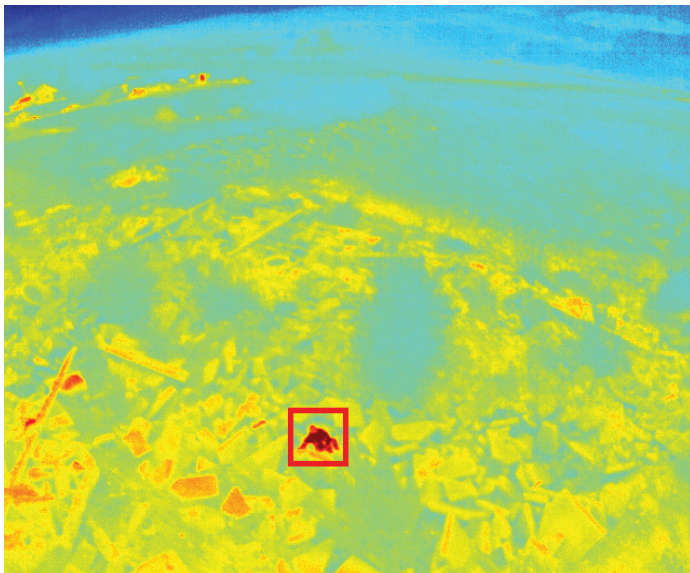**Figure 8.** Automated victim detection on a rubble field in an urban search and rescue operation (source: ICARUS).

Figure 9. Victim in the water being localized by outdoor multi-rotor aircraft.

Figure 10. Collaborative indoor victim search (source: ICARUS).

Figure 11. Collaborative maritime victim search and rescue operation involving aerial and maritime platforms (source: ICARUS).

7.4. Carrier

During any search and rescue operation, many assets need to be deployed as quickly as possible. This mission profile shows how collaborative robotic agents can help by acting as carrier platforms for small assets and equipment and also for other unmanned systems. As an example, the large unmanned ground vehicle acts as a carrier platform for the small-unmanned ground vehicle and both the ROAZ II and the U-ranger act as a carrier for the rescue capsules. They not only enable the cargo to be transported to the destination, without any extra burden to the human relief workers, but also act as deployment systems for the smaller unmanned systems.

As an example, **Figure 12** shows how the large unmanned ground vehicle deploys the small-unmanned ground vehicle on the top of a building, whereas **Figure 13** shows how the rescue capsules are deployed from an ICARUS unmanned surface vehicle.

Figure 12. ICARUS large unmanned ground vehicle deploying the small unmanned ground vehicle on the top of a building (source: ICARUS).

Figure 13. ICARUS unmanned surface vehicle deploying the unmanned rescue capsule (source: ICARUS).

7.5. Communications relay

In the event of a large crisis, previously existing communication infrastructure is often broken or at least severely damaged. However, communication is crucial for having coordinated response operations. Collaborative unmanned systems can act as communication relay tools to extend the communication range over large distances. Of course, the assets which are most useful for this are the aerial tools, as they can provide line-of-sight communication relay over large distances. In the ICARUS project, an ad-hoc link-hopping network was developed, as detailed in Chapter 7 of this book, which allows to extend any communication link while the ICARUS aerial platforms are in the air. This allows the fixed wing aircraft and the outdoor multi-rotor aircraft to act as communication relays for the ground and marine rescue teams.

7.6. Air, sea, ground cooperation during euRathlon 2015

Only seldom, rescue operations have to be performed which span three domains (air, ground, marine). However, the Tohoku earthquake and subsequent Fukushima disaster showed that response protocols were ill prepared to approach such multi-domain crises. Therefore, the euRathlon challenge focussed specifically on this problem. In brief, the mission imposed by the euRathlon challenge consists of detection of, after a Fukushima-like incident, missing workers under water, outside on the ground and inside in a semi-demolished reactor building. Full information of the concept of operation is available online [23]. These search and rescue operations require the simultaneous and coordinated deployment of unmanned aerial, ground and underwater vehicles, gathering environmental data and performing real-time identification of critical hazards in a nuclear accident.

The ICARUS team deployed for this purpose five robotic systems:

- The outdoor multi-rotor was first deployed to search for the best route for the ground robots to reach the open entrance to the building. Then, it mapped the area in the RGB, gray and thermal spectrum. Finally, it performed real-time detection and localization of missing workers, leaks, etc.

- The Teodor UGV was used as carrier platform for the small UGV and for outdoor 3D mapping.

- The small UGV was used for indoor detection and localization of missing workers, leaks, etc.

- The ROAZ II vehicle was used as a carrier platform for the MARES unmanned underwater vehicle.

- The MARES unmanned underwater vehicle was used for underwater detection and localization of missing workers, leaks, etc.

Even though behaviours specific to euRathlon, such as opening valves were not originally considered in the ICARUS concept of operations, they were easily integrated as a proof of the flexibility of the followed approach towards interoperability. Thanks to the different levels of interoperability and automation, the specialized operator could take over at this point and tele-operate the system to finish the mission.

The following figures show the ICARUS team operating in the euRathlon scenario. **Figure 14** shows the ICARUS multi-rotor during his flight around the building, while **Figure 15** illustrates the Teodor UGV carrying the small UGV during the euRathlon challenge.

Figure 14. ICARUS rotorcraft during the euRathlon challenge (source: ICARUS).

Figure 15. Teodor and small UGV during the euRathlon challenge (source: ICARUS).
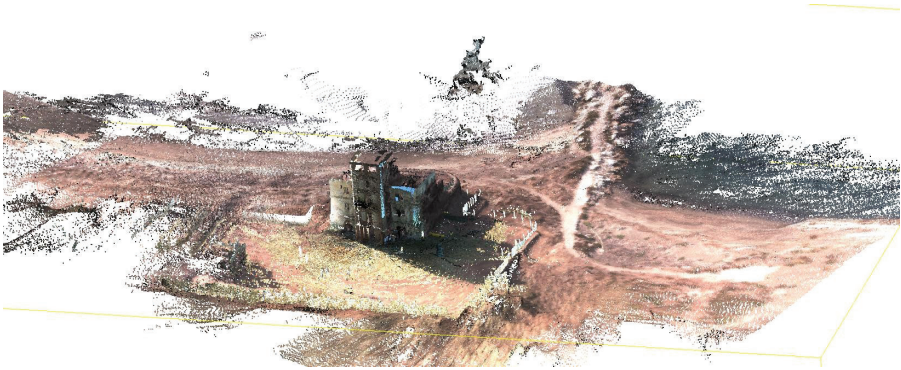
Figure 16. 3D map of the crisis area, obtained by combining 3D maps from the aerial and ground platforms (source: ICARUS).

Figure 16 shows the outcome of combining 3D maps obtained from the outdoor multi-rotor and ground platforms during the euRathlon challenge.

8. Conclusions

The work described in this chapter intended to integrate unmanned air, ground and sea vehicles developed by the different ICARUS partners into a heterogeneous fleet, collaborating as a coordinated, seamlessly-integrated team. A strong effort was devoted to appraise the existing body of work in standardization of multi-robot systems. Given the particular requirements of ICARUS, emphasis was placed on initiatives considering multiple domains (air, ground and sea). Likewise, given the platforms used in ICARUS, standards and methods applicable to smaller and lightweight platforms were prioritized. There have been several initiatives addressing both issues. However, harmonization of them is not yet a fact. There is still the need for a single multi-domain standard for interoperability, easily adaptable to both large and small systems. The contribution of the ICARUS project focused on the selection of the most appropriate existing initiative (JAUS), the evaluation of its application to multi-robot Search and Rescue missions, the elaboration of recommendations for improvements, the adaptation of all ICARUS robots and the demonstration of the ICARUS interoperable and heterogeneous team in three large-scale demonstration, exploring multi-robot cooperation and real-time centralized supervision and planning of an heterogeneous team.

Acknowledgements

Author details

Daniel Serrano López[1]*, German Moreno[1], Jose Cordero[2], Jose Sanchez[2], Shashank Govindaraj[3], Mario Monteiro Marques[4], Victor Lobo[4], Stefano Fioravanti[5], Alberto Grati[5], Konrad Rudin[6], Massimo Tosa[7], Anibal Matos[8], Andre Dias[8], Alfredo Martins[8], Janusz Bedkowski[9], Haris Balta[10] and Geert De Cubber[10]

*Address all correspondence to: daniel.serrano@eurecat.org

1 Eurecat Technology Center, Av. Universitat Autònoma, Cerdanyola del Vallès, Barcelona, Spain

2 Integrasys SA, Madrid, Spain

3 Space Applications Services NV, Zaventem, Belgium

4 Escola Naval, Rua Base Naval de Lisboa, Almada, Portugal

5 NATO STO Centre for Maritime Research and Experimentation, La Spezia, Italy

6 Eidgenoessische Technische Hochschüle Zürich, Zürich, Switzerland

7 Technische Univeristät Kaiserslautern, Kaiserslautern, Germany

8 INESC TEC – Instituto de Engenharia de Sistemas e Computadores, Tecnologia e Ciência and Faculdade de Engenharia da Universidade do Porto, Campus da FEUP, Porto, Portugal

9 Instytut Maszyn Matematycznych, Warszawa, Poland

10 Department of Mechanical Engineering, Royal Military Academy of Belgium, Brussels, Belgium

References

[1] Freeman, E., Robson, E., Sierra, K., & Bates, B. (2004). Head First design patterns. Sebastopol, CA: O'Reilly

[2] Serrano D, Chrobocinski P, De Cubber G, Moore D, Leventakis G, Govindaraj S. ICARUS and DARIUS approaches towards interoperability. In: 8th IARP Workshop on Robotics for Risky Environment; Lisbon, Portugal; IARP January 2015, pp. 1-12

[3] Serrano D. Key initiatives for interoperability through standardization. Applicability to small unmanned vehicles. In: The NATO STO Lecture Series SCI-271. 2015. NATO Science and Technical Organization – Educational Notes. DOI: 10.14339/STO-EN-SCI-271-01-pdf

[4] Studer R, Benjamins VR, Fensel D. Knowledge engineering: Principles and methods. Data and Knowledge Engineering. 1998;**25**:161-197

[5] Prestes E, Carbonera JL, Fiorini SR, Jorge VAM, Abel M, Madhavan R, Locoro A, Goncalves P, Barreto ME, Habib M, Chibani A, Gérard S, Amirat Y, Schlenoff C. Towards a core ontology for robotics and automation. Robotics and Autonomous Systems. 2013;**61**(11):1193-1204. DOI: http://dx.doi.org/10.1016/j.robot.2013.04.005

[6]   NATO. STANAG 4586, Edition No 2, Standard Interfaces of UAV Control System (UCS) for NATO UAV Interoperability – 2007

[7]   Andreas Birk and Max Pfingsthorn. 2006. A HMI supporting adjustable autonomy of rescue robots. In RoboCup 2005, Ansgar Bredenfeld, Adam Jacoff, Itsuki Noda, and Yasutake Takahashi (Eds.). Springer-Verlag, Berlin, Heidelberg 255-266

[8]   Goodrich M, Olsen D, Crandall JW, Palmer TJ. Experiments in adjustable autonomy. In: IEEE Xplore Conference: Systems, Man, and Cybernetics, 2001 IEEE International Conference on, Volume: 3

[9]   R. Parasuraman, T. B. Sheridan and C. D. Wickens, "A model for types and levels of human interaction with automation," in IEEE Transactions on Systems, Man, and Cybernetics - Part A: Systems and Humans, vol. 30, no. 3, pp. 286-297, May 2000. DOI: 10.1109/3468.844354

[10] Lacroix S, Alami R, Lemaire T, Hattenberger G, Gancet J. Decision making in multi-UAVs systems: Architecture and llgorithms. In: Ollero A, Maza I, editors. Multiple Heterogenous Unmanned Aerial Vehicles. Springer Tracts in Advanced Robotics ed. Berlin: Springer; 2007. pp. 15-48. DOI: 10.1007/978-3-540-73958-6_2

[11] Farinelli A, Iocchi L, Nardi D. Multirobot systems: A classification focused on coordination. IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics). 2004;**34**(5):2015-2028

[12] ISO. ISO Standards Website – 2017 [Internet]. Available from: http://www.iso.org/iso/home/standards.htm

[13] Pedersen J. Interoperability Standard Analysis (ISA). Standards Committee of the National Defense Industry Association (NDIA) Robotics Division; 2007

[14] NATO. STANAG 4586, Edition No 3, Standard Interfaces of UAV Control System (UCS) for NATO UAV Interoperability – 2012

[15] Incze ML, Sideleau SR, Gagner C, Pippin CA. Communication and collaboration of heterogeneous unmanned systems using the joint architecture for Unmanned Systems (JAUS) standards. In: OCEANS 2015; 18-21 may 2015: Genova, Italy. 2015. pp. 1-6

[16] Rowe S, Wagner CR. An introduction to the joint architecture for unmanned systems. Technical Report, Ann Arbor, MI, USA. Cybernet Systems Coorporation

[17] SAE International. JAUS Core Service Set [Internet]. 2008. Available from: http://standards.sae.org/as5710/

[18] SAE International. JAUS Mobility Service Set [Internet]. 2009. Available from: http://standards.sae.org/as6009/

[19] SAE International. JAUS Environment Sensing Service Set [Internet]. 2015. Available from: http://standards.sae.org/as6060/

[20] SAE International. JAUS Manipulator Service Set [Internet]. 2011. Available from: http://
standards.sae.org/as6057/

[21] QGroundControl. MAVLink Micro Air Vehicle Communication Protocol [Internet].
2016. Available from: http://qgroundcontrol.org/mavlink/start

[22] Balta, H., Bedkowski, J., Govindaraj, S., Majek, K., Musialik, P., Serrano, D., Alexis,
K., Siegwart, R. and De Cubber, G. (2017), Integrated Data Management for a Fleet of
Search-and-rescue Robots. J. Field Robotics, **34**: 539-582. DOI:10.1002/rob.21651

[23] Ferri G, Ferreira F, Djapic V, Petillot Y, Franco MP, Winfield A. The euRathlon 2015
grand challenge: The first outdoor multi-domain search and rescue robotics competi-
tion—A marine perspective. Marine Technology Society Journal. 2016;**50**(4):81-97

Tactical Communications for Cooperative SAR Robot Missions

José Manuel Sanchez, José Cordero,
Hafeez M. Chaudhary, Bart Sheers and
Yudani Riobó

Abstract

This chapter describes how the ICARUS communications (COM) team defined, developed and implemented an integrated wireless communication system to ensure an interoperable and dependable networking capability for both human and robotic search and rescue field teams and crisis managers. It starts explaining the analysis of the requirements and the context of the project, the existing solutions and the design of the ICARUS communication system to fulfil all the project needs. Next, it addresses the implementation process of the required networking capabilities, and finally, it explains how the ICARUS communication system and associated tools have been integrated in the overall mission systems and have been validated to provide reliable communications for real-time information sharing during search and rescue operations in hostile conditions.

Keywords: communications, mesh, contention, optimisation, middleware, propagation

1. Introduction

First responders' communications (COM) have become a key concern in large crisis events which involve numerous organisations, human responders and an increasing amount of unmanned systems which offer precious but bandwidth-hungry situational awareness capabilities.

The ICARUS team in charge of developing the COM system — lead by INTEGRASYS with contributions from RMA and QUOBIS — has designed, implemented and tested in real-life conditions an integrated multi-radio tactical network able to fulfil the new demands of

cooperating high-tech search and rescue teams acting in incident spots. The ICARUS network offers interoperable and reliable communications with particular consideration of cooperative unmanned air, sea and land vehicles.

In this chapter, we provide a description of the different phases. Starting with requirements collected from high-level mission managers and specific platform operators, we describe the key design decisions taken by the COM team to follow with implementation details and finalising with the COM system results obtained during the different trials conducted by the project.

2. Communication scenarios and requirements

Proper communication systems are needed to ensure the networking capability that allows SAR team members (robots and humans) and operations managers to share real-time information under the hostile operating conditions characterising disaster-relief operations [1–3]. These conditions mandate the use of wireless communication technologies to support the inherent mobility nature of operations [4, 5].

Figure 1 depicts the general information exchanges occurring in typical disaster-relief operations where multiple SAR teams are actuating. An entity named on-site operations coordination centre (OSOCC) acts as the central coordination centre for all operations and is placed close to the disaster zone. First, area reduction and sectorisation tasks are performed by the OSOC to quickly identify and analyse priority actuation areas so as to allocate specific sectors
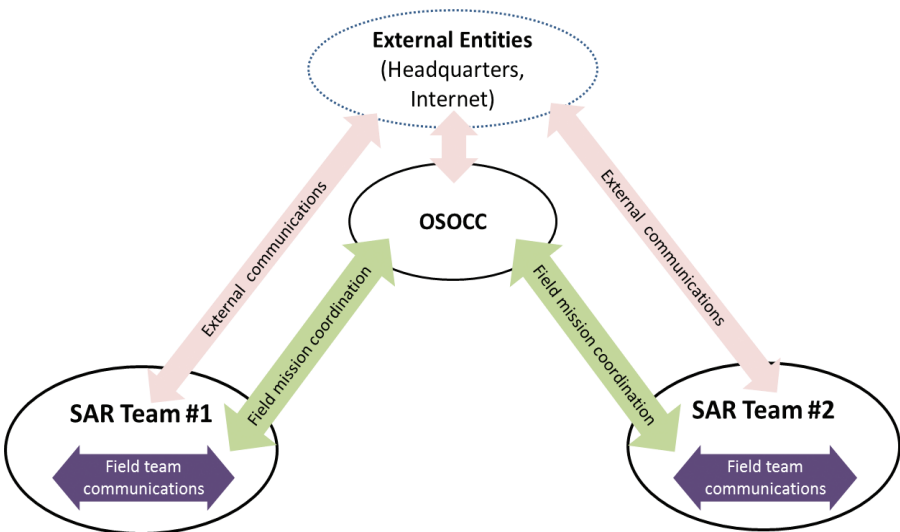

Figure 1. High-level communications in ICARUS. (Source: ICARUS).

to available SAR teams. These initial planning activities are likely done by the OSOCC with the support of unmanned assets of SAR team temporarily collocated with the OSOCC.

The SAR teams are groups of first responders equipped with unmanned vehicles that perform SAR operations in an allocated area. They use team-internal communications (labelled Field Team Communications in the figure) to perform their activities, in particular sharing sensor information captured by human or robotic responders and commanding unmanned vehicles from control stations. The SAR team activities are supervised and coordinated by the OSSOC using field mission communications, which serve, for example, to report about rescued victims, current team-members' location, new actuation areas, etc. Both the OSOCC and the SAR teams may make use of external communications with distant entities, such as agencies headquarters for logistic coordination, or data servers providing background or newly acquired information about the disaster area.

Building upon the reference ICARUS communication scenario described above, the ICARUS COM team worked in closed cooperation with other project teams to gather a list of relevant requirements to guide the COM system design and further implementation. Feedback obtained from end-users (SAR organisations) participating in the project either as partners or as end user board (EUB) experts was used to compile a list of essential high-level requirements, which is shown in **Table 1**. From this list, we highlight in particular the need of using non-reserved spectrum for the operations, due to the likely impossibility of using pre-existing local communication infrastructures and coordinating with the national spectrum regulation in the early phases of a crisis event.

Furthermore, in collaboration with the different project teams in charge of defining overall user requirements, providing unmanned platforms and developing the interoperable Command & Control (C2) tools, an extended view of the communication architecture was elaborated together with a list of quantitative performance target for the ICARUS COM system, based on expected equipment sizing of future SAR teams.

End users COM requirements	
Description	Level
Affordable solution	Mandatory
Support sectorisation and SAR operations	Mandatory
QoS support	Mandatory
Over the air security	Mandatory
Ad hoc capability	Mandatory
Unlicensed spectrum operation	Mandatory
Easy and uniform management and control	Mandatory
High temporal and spatial availability	Mandatory
Interoperability with existing networks	Desirable

Table 1. ICARUS communication requirements stated from SAR end users.

Figure 2 shows the refined view of the communications architecture where the field team communications within a SAR team operation area are populated with different entities and networking segments. This architecture constitutes the reference ICARUS communication model and reflects the typical command and control architecture of future SAR missions making use of ICARUS tools. Each SAR team has a base of operations (BoO) entity which coordinates different squads, namely a group of human and robot responders working in a specific spot within the assigned SAT team area. Making use of a Squad coordination network, each squad operates its unmanned assets through a robot command and control (RC2) station, which additionally serves as a base station for human communications, either voice-based or message-based. The BoO receives mission guidance and reports mission status to the OSOCC through the team coordination network segment and at the same time executes the assigned team mission in coordination with the different squads through the team coordination network segment. In **Figure 2**, it can be seen several COM management entities, residing on the different system entities forming a hierarchical structure that will cooperatively perform all management and control functions on underlying COM resources to allow first responders and their tools to be smoothly interconnected during operations. The network segmentation shown in **Figure 2** does not assume a corresponding physical segmentation in terms of frequency channels, link-level networks or IP-level networks; it is rather a logical organisation resembling the working structure of teams.

Table 2 gathers the list of key performance targets for the ICARUS COM system elaborated in cooperation with end user organisations, unmanned platform providers and C2 system
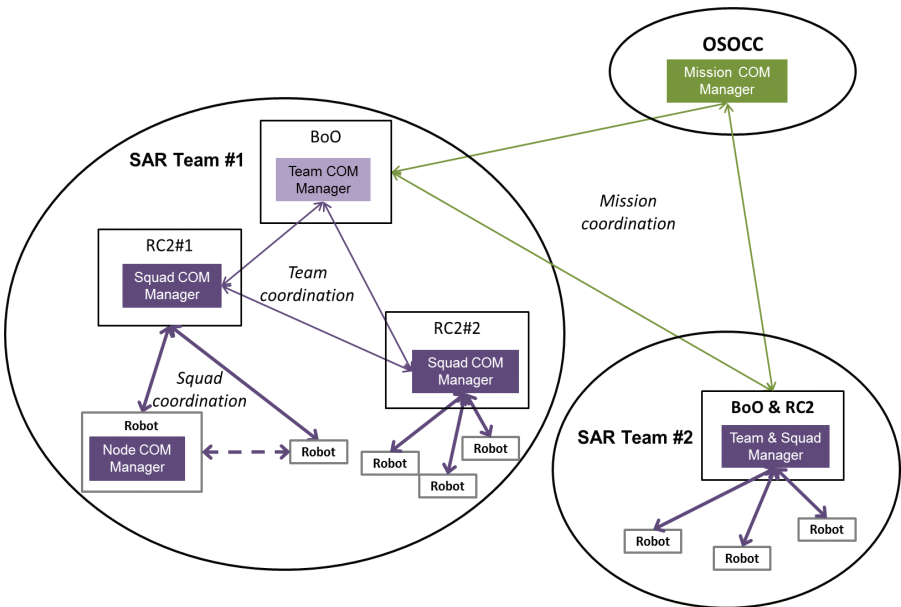

Figure 2. High-level communication segments in ICARUS. (Source: ICARUS).

Quantitative requirements

Description	Value	Scenario	Level
Maximum range	10 Km	Sectorisation	Mandatory
	1.5 Km	Outdoor SAR	
	500	Indoor SAR	
	100	Rubble SAR	
Max. squad nodes	5	All	Mandatory
Max. squads	3	All	Mandatory
Critical payload	Video feed (500 Kbps)	SAR	Mandatory
	Exoskeleton (250 Kbps)		
Peak capacity	100 Kbps@backhaul	Sectorisation	Desirable
	670 Kbps@spot		
	100 Kbps@backhaul	SAR	
	670 Kbps@spot		
Maximum platform mobility	100 Km/h	Sectorisation	Desirable
Squad handover time	30 s	SAR	Desirable

Table 2. ICARUS communication performance targets.

providers. The reference team networking scenario consists of several interconnected squads operating in cell areas with a maximum radius of 1500 m and five nodes, including a R2C station, which should be able to transition across squads within a limited time. A mix of synchronous/asynchronous application traffic is transferred within squads, between the squads and with the OSOCC. The estimated peak capacities include typical video, voice and Telemetry/Telecommand (TM/TC) feeds.

3. Pre-existing solutions and design decisions

Providing reliable wireless connectivity during disaster relief presents a significant challenge. For robust and effective disaster response, mesh wireless networking technology presents a solution to create adaptive network in emergency scenarios in which support infrastructure is either scare or non-existent [5–11]. A flexible mesh network architecture that provides a common networking platform for heterogeneous multi-operator networks, for operation in case of emergencies, is proposed in Ref. [5]. In Ref. [12], the authors have proposed an approach to establish a wireless access network on-the-fly in a disaster-hit area relying on the surviving access points or base stations, and end-user mobile devices. Similar works also appear in Refs. [13, 14]. An ad hoc networking solution is proposed in Ref. [15] to aid emergency response relying on WiFi-Direct enabled consumer electronic

devices such as smartphones, tablets, and laptops. An integrated communication system is proposed in Ref. [16] comprising heterogeneous wireless networks to facilitate communication and information collection on the disaster site. Based on WiMAX technology, without fixed access point, an ad hoc networking solution is proposed in Ref. [17] using UAV relays to realise a backbone network during emergency situations. Similar concept is proposed in Ref. [18] using IEEE802.11s. A recent work in Ref. [19] employs dual wireless access technology for robotic assisted SAR operations–one technology to provide a long-range, single-hop, low bandwidth network for coordination and control of the robotic devices and second technology for short-range, multi-hop, high-bandwidth network for sensor data collection. Ref. [20] proposes a framework for modelling and simulating the communication networks and examining the ways in which availability, quality of the communication links, and the user engagement affect the overall delays in disaster management and relief. Leveraging the latest advances in wireless networking and unmanned robotic devices, Ref. [21] proposes a framework and network architecture for effective disaster prediction, assessment and relief.

As we have seen in the previous sections, the ICARUS SAR scenario demands QoS-enforced wireless communications for different types of nodes (robots and stations) spread over a relatively large area in order to provide proper throughput, latency and reliability for the different applications needed to support the missions. Furthermore, future robotic C2 systems enabling higher autonomy – for example, those supported by the JAUS framework selected in ICARUS – will dynamically use centralised and decentralised algorithms [22], demanding from the communications layer the ability to have a flexible balance of the uplink (transmission) and downlink (reception) capacity of network nodes.

Previous research or demonstration activities dealing with a cooperative robotic scenario similar to ICARUS have commonly deployed different technologies, either standards-based such as PMR (Professional Mobile Radio, e.g. TETRA), WLAN (802.11 family of standards), WPAN (802.15.4 family) and WMAN (802.16 family), or proprietary-based solutions in licensed or unlicensed spectrum; complemented with public services such as 3G/4G or WiMax, in case these were available at the operations area. As no single communication technology is able to satisfy the varied set of requirements usually demanded by the users, a combination of several datalinks is recurrently used to provide the communication service.

In order to facilitate the selection of the most appropriate datalink technologies for ICARUS, a reduced set of operational and technology challenges to be solved in order to provide a proper, real-world communication solution for the posed scenario was defined in cooperation with end users. These challenges are shown on the left side of **Table 3**, followed at the right side by the corresponding approaches taken by the COM team to address them building upon existing datalinks.

While the various datalink technologies surveyed present rather different features and capabilities, the COM team focused on the specific set of wanted characteristics that served most to solve the challenges identified. As an example, in the following, we list some of the key wanted features at the datalink level.

Category	Challenge	Response
Cross-cutting, operations & management	Heterogeneity of robotics platforms and operation environments	Variety of COM options (HW, radio bands, datalink options) offered in uniform way
	Minimal configuration and integration effort for robot platforms and C2I system providers	Custom application MW traffic processing in COM
		Single interface for COM management collocated with robot fleet management
	Guarantee robustness and real-time performance with affordable hardware	Reliability enforcement via software
	Need to have dynamic allocation of robots to C2I stations (teams)	Change of robot-to-RC2 allocations via expedite software reconfiguration
Datalink technology	Maintain reliable connectivity in unlicensed spectrum	Cognitive radio, reduced bandwidths, fast channel switching, channel/band aggregation
	Achieve long ranges in unlicensed bands	Relays, proper bands/channels and transceivers
	Maintain shared link/flow status in harsh, highly changing network conditions	Network timing, synchronisation and recovery mechanisms
	Avoid network congestion	Application adaptation, local safeguards, global admission control with pervasive performance monitoring

Table 3. Key communication challenges in robotic SAR scenarios and the ICARUS responses.

- Dynamic channel selection and frequency hopping to improve reliability in unlicensed spectrum where multiple competing networks may exist.

- Multi-hop capable datalinks or the lowest possible spectrum bands (e.g. 433 and 868 MHz) looking for favourable propagation conditions to achieve long ranges in unlicensed spectrum. Both approaches come at the expense of reduced bandwidth.

- Modulations resilient to non-line-of-sight conditions, link diversity solutions (e.g. link meshing or MIMO antennas), and rate and transmission power control to cope with variable link conditions experienced by mobile nodes, subject, for example, to blocking obstacles.

- Proper QoS techniques to avoid network congestion while guaranteeing performance for the individual flows generated at the different nodes. QoS can be guaranteed on a deterministic basis with a channel access scheme based (at least partially) on time-slots allocation, which requires time synchronisation between network nodes and may add significant control traffic overhead if frequent reallocation of capacities is needed. QoS performance highly depends on network topology, and some datalink technologies (e.g. those used in sensor networks) are designed for specific application cases (e.g. cluster-tree topologies), which limits usability in the ICARUS scenarios.

These considerations on datalink technologies must be traded-off with wanted high-level system features and overall non-functional requirements, as stated in **Table 3**, observing at all times the need to have an affordable solution.

From a system level perspective, ICARUS C2 applications are operating upon the JAUS middleware, assuming transparent IP connectivity between the different end nodes. Therefore, solutions are needed to integrate the different datalink technologies and link-layer subnetworks in an interoperable IP addressing space and to properly propagate QoS settings for different exchanges from the middleware level down to the datalink layers. Some datalink technologies are not IP-capable due to resource constrains of the node platforms (e.g. sensors), which adds further difficulty leading to the implementation of IP gateways which must properly translate all needed IP protocols to the link layer. On the other hand, a specific requirement is the ability to transfer the control of robots between different stations operation potentially in different areas, so roaming over different network segments would be required. There are generic solutions at the IP level which provide multi-homing and mobility support but are rarely applied in ICARUS-like scenarios due to the effort needed to synchronise mechanisms at IP-level with those needed at the underlying link-level for the several datalink technologies used.

Having all of the above considerations in mind a detailed comparative study of available solutions was made, resulting in the final selection of the following technologies:

- ETSI digital mobile radio (DMR) datalink [23] for long-range low-rate communications between control stations and robots. Aiming at an open and affordable hardware implementation using commercial components, a Tier-2 direct-mode operation is selected with multiple coding options to avail of capacity versus range flexibility. This is extended with software-based functions allowing valuable services such as node discovery and capacity management. The latter allows to accommodate different traffic arrival patters latency requirements procuring maximum network utilisation.

- IEEE 802.11n network [24] with meshed multi-hop support to interconnect the different squads, teams and the OSOCC. Building upon commercial transceivers, extended management and control functions based on open Linux-based software are identified to achieve high performance in ICARUS environments, based on the smart handling of channel, power/rate, CSMA and EDCA parameters. Spectrum-level functions such as channel selection and power control are supported by cognitive radio techniques [25], aiming at operation with minimum interference and maximum spatial reusability conditions. The use of such cognitive radio features in disaster response networks offers opportunities to adapt communication links to the various changes in the operating environment and thereby enhance the performance of the communication network [26].

The proper integration, extension and smart utilisation of the two types of datalink selected are expected to provide the concrete responses to the ICARUS COM challenges found at the right side of **Table 3**, which form the key design aspects of the ICARUS COM solution.

4. The implemented ICARUS COM system

4.1. Interoperability, performance and manageability functions

The ICARUS COM team approach to implement the required networking capability for SAR missions is to implement key software-based functions upon well-established, commercial datalink technologies offering managed performance levels with enough predictability. The combined set of functions will ensure instant interoperability among the variety of unmanned vehicles, personal devices and control stations and will enable performance optimisation by adapting to changing conditions due, for example, to nodes mobility, propagation environment, external interference or evolving mission needs.

The implemented ICARUS COM functions are grouped in three different areas: (a) radio resources management, (b) IP protocol addressing and routing management and (c) overall management and control (M&C).

At radio resources level, ICARUS implements a distributed cognitive radio capability to allow dynamic channel selection (frequency and width) over different unlicensed spectrum bands – 433 MHz, 870 MHz, 2.4 GHz and 5 GHz – for the whole set of datalinks and network segments used in the system. An innovative combination of raw spectrum monitoring with physical and link layer measurements from network devices provides a global view for channel selection as well as a per-link view to quickly detect problems and take proper correction actions; procuring at the same time implementation of required regulation rules to access given spectrum bands.

At IP protocol layer, a single virtual IP network is offered to applications building upon native operating system tools. Rather than providing a single IP to each system platform (robot or control station), an IP subnet sized for six different addresses is allocated, so that different physical nodes corresponding the same platform (e.g. main computer and standalone cameras on-board the same vehicle) can access to the ICARUS communication capability available on a dedicate COM computer hosting the COM software and datalinks. Proper routing functions ensure that unicast and multicast application traffic running over the virtual IP network smoothly traverses multiple wired and wireless link-layer segments.

All of the IP traffic handled in the ICARUS is QoS marked so that proper processing can be done first within the IP stacks of the system nodes and further within the operating datalink layer. In SAR communications, it is imperative to be able to handle different application flows with different QoS giving priority to certain types of data. Based on the defined requirements in Section 2, a number of traffic classes have been defined in the ICARUS COM system, which are shown in **Table 4** detailing the differentiating characteristics and typical application flows making use of them.

At overall M&C level, a coordinated set of managers and controllers' modules is designed to handle the traffic generated from the JAUS application middleware to be properly

QoS classes						
	Access priority	Delay enforcement	Throughput enforcement	Reliability enforcement	Pre-emption	Flow examples
Critical	First	High	High	Yes	Yes	Network M&C
						Vehicle TM/TC
						Exoskeleton TM/TC
						Robotic MW signalling
Real time	Second	Medium	Medium	Yes	Yes	Primary and secondary real time imaging
			High			
Best effort	No	No	No	Yes	No	Sensor data downloading
						Secondary real-time imaging

Table 4. ICARUS communication QoS classes.

transferred through the underlying datalinks. To that end, the COM layer implements automated JAUS traffic identification and subsequent QoS allocation based on a set of pre-defined and run-time reconfigurable rules so that no change is needed on existing applications to benefit from the managed communication capacity of ICARUS so that custom application MW traffic processing in COM easy-to-use software interfacing mechanism will be provided within the middleware itself. In addition to passing data units, applications will use the interface to select applicable QoS parameters, while the COM layer will provide relevant information about connectivity (e.g. reachability of other nodes, capacity limits, etc.) using the same naming rules used by the middleware. In this way, control algorithms can conveniently include communication status information to take better decisions.

4.2. The architecture of the ICARUS COM nodes

The set of COM functions briefly introduced in the previous section is implemented in the form of software modules residing in computing nodes associated with the different system entities, namely unmanned vehicles and corresponding control stations, personal devices and mission coordination stations. The various software modules need to efficiently interface with each other — either within the same or over different platforms nodes — to undertake different control, data or management functions. In order to facilitate the implementation of the ICARUS COM system as well to allow for future extensibility, well-structured and formal mechanisms were defined to model, develop and deploy the different ICARUS software modules. The set of core modules supporting this mechanism and implementing essential system

functions is known as the ICARUS COM middleware (COMMW). The COMMW enables the implementation of cooperative and specialised management and control functions and has therefore been a key piece enabling interoperable and resilient tactical communications in the ICARUS scenario of crisis response operations covering air/sea/land portable and mobile nodes.

Figure 3 represents the key COM modules residing in the four different nodes forming a single robot control setup. Two of them (APPNODEs) represent the main computers aboard a robot and at the RC2 station hosting all the software needed for controlling and supervising the platform and its payload sensors. The other two (COMNODEs) are small computers linked through Ethernet connection to their corresponding application nodes acting as data routers providing access to the ICARUS wireless network. In the case of the RC2 station, management and control interfaces are also established between given entities at communication and application levels for overall monitoring and control of mission communications during operations. In the figure, there can be easily identified the different layers constituting the ICARUS COMMW.

The COMMW has been implemented on open, Linux-based embedded computing platforms with proper kernel and user-space extensions enabling an overall optimisation of the network stack, including the queuing components present in the system data path, which may largely affect throughput and latency of applications. **Figure 4** below shows the final aspect of the assembled COM computer mounted aboard the so called LUGV (Large Unmanned Ground Vehicle) ICARUS robot.
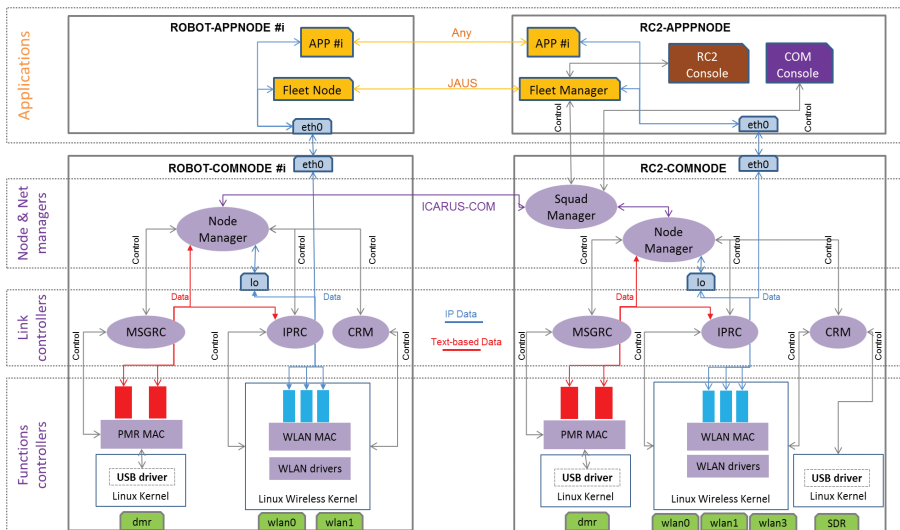

Figure 3. High-level communication segments in ICARUS. (Source: ICARUS).

Figure 4. SUGV COM box and set of antennas used in various missions. (Source: ICARUS).

The COMMW framework seamlessly integrates and jointly manages both WLAN and DMR datalinks according to dynamic mission conditions and evolving requirements. In the following sections, we describe the key datalink-specific functions implemented.

4.3. DMR datalink implementation

The DMR datalink technology standardised by ETSI provides long range coverage (typically beyond 5 km in open areas) and can handle both voice and low-rate data. The so-called soft-DMR modem implemented in ICARUS [27] enables adaptation of key transmission parameters — coding rate, delivery mode, channel access mode and transmission power — on a per-destination basis, according to QoS requirements (**Table 4**) of the currently handled application data. As ICARUS extensions to the DMR Tier-2 technology, a node discovery service and a capacity management protocol (allowing allocation of throughput levels per node) were implemented to strength the networking aspects of DMR. All these characteristics make the soft-DMR well suited for networked tactical and mission critical applications.

The following **Figure 5** shows the final DMR modem board implemented together with an average ballpoint pen for size comparison purposes.
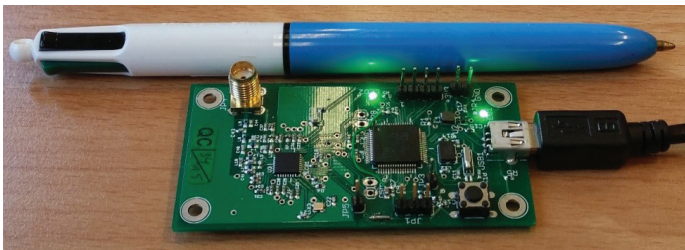

Figure 5. ICARUS DMR hardware transceiver. (Source: ICARUS).

4.4. WLAN datalink implementation

ICARUS WLAN datalinks are based on 802.11n commercial transceivers with 2 × 2 MIMO antenna configuration which was assessed as a fair setup to operate in the variety of radio propagation conditions existing in ICARUS missions. All used transceivers are equipped with an Atheros dual-band chipset supported by the Ath9k Linux driver, which is the common basis to develop low-level ICARUS extensions. Full-mesh capability spanning multiple frequency channels is provided through the 802.11s Linux implementation, properly configured to allow a smooth behaviour of mesh peering and routing algorithms given the particular mobility and radio link conditions expected for ICARUS nodes.

Specific functions deployed in Kernel space for performance reasons allow the fine control of key system parameters affecting the overall network performance — particularly range and throughput — which are optimised in real-time according to predefined and reconfigurable operator policies. These parameters refer to three distinct areas:

- At radio link level, the controlled parameters are: radio bands and channels frequencies and widths; transmitted power, rate control policy, frame retry policy and waveform mode (e.g. 11b, 11g, or 11n). Legacy waveforms are eventually used for nodes under particularly disadvantaged radio conditions, for example, located at long distances or in indoor.

- At channel access level, the controlled parameters are per-class EDCA contention parameters and the CSMA carrier sense level.

- At mesh protocols levels, the controlled parameters are timers and counters associated with paths and peers' discovery and association protocols; and to the configuration of root and gateway nodes.

4.5. Operational management

In parallel to the implementation of COM managers and controller modules, the ICARUS COM team worked in the development of a convenient set of tools to ease the tasks of operators responsible for communications during the different mission phases (planning, deployment, operation) aiming at simplified and fast manual interventions while having proper information and tools at all times to fine-tune key parameters affecting the performance of the overall network and specific links.

There are two different toolsets offered to network operators. The first one is a configuration tool based on a structured data model which allows to setup the overall node configuration based on capacity allocation targets for both locally-generated and relayed traffic; differentiating among individual application flows and supporting latency, reliability and security requirements in addition to throughput. Operators are provided with a set of utilities for guidance on setting the different configuration parameters. Some of the settings will be subject to dynamic changes during mission execution.

The second one is a rich graphical environment named COM console (COMCON) conceived to support planning, supervision and optimisation of the integrated multi-radio ICARUS

network, combining simulation features with real-time monitoring and control capabilities. In both simulation and real-time modes, the COMCON tool acts as a visualisation and control frontend for the COMMW modules. The COMCON tool is able to represent with high-fidelity the time behaviour of the ICARUS network with fine-grained view and control of a number of interrelated physical or system factors, which influence the performance of specific links and the overall network.

At planning phase, the COMCON tool accurately characterises COM components, propagation environments, RF interference and vehicles platforms in order to assess global network performance over wide operation areas; as well as the performance of individual terminals along given mission routes. This allows in particular to take proper decisions on radio bands and channels, antennas pattern/polarisation and transceiver features for every node in the network. Furthermore, the eventual need and location of network relays can be assessed. The tool includes propagation models for indoor, rubble and sea environments in UHF/2.4 GHz/5 GHz bands; as well as protocol models of 802.11 mesh networks enabling informed planning of CSMA-related parameters and reliable estimation of throughput performance. **Figure 6** exemplifies a mission modelled in the COMCON tool where the different links and antenna coverages of networking nodes are calculated and verified during mission planning in an interactive 3D Earth Globe visualisation interface.

At operations phase, the COMCON features a centralised monitoring of all key parameters affecting the network performance, allowing to mitigate coverage and throughput problems by timely reconfiguration and eventual reallocation of nodes. **Figure 7** shows an example of a real mission monitoring display offering connectivity as well as link performance information to the operator.

Some optimisation actions of limited impact are performed automatically by the COMMW stacks, while some others of wider scope require human operator intervention to decide the best solution given the current mission conditions. Of special relevance to network operators
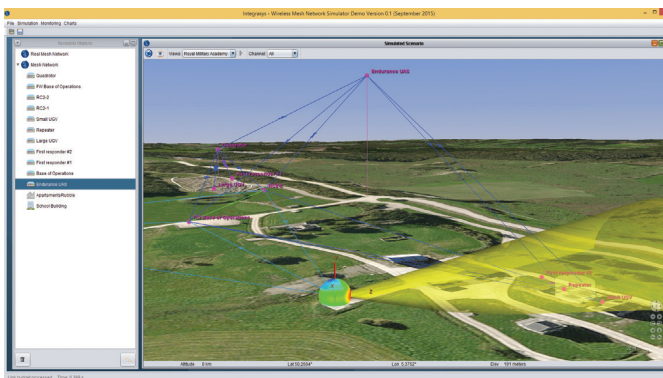

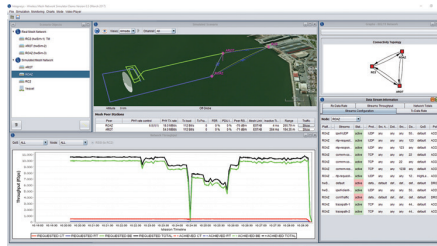**Figure 6.** ICARUS COM console used in mission planning. (Source: ICARUS).

Figure 7. ICARUS COM console in mission operations. (Source: ICARUS).

is the ability of the combined COMMW software and COMCON tool to determine the likely reason of detected traffic losses, leading to different corrections. The traffic losses are classified in four different groups:

- Collisions, which can be solved by forcing RTS/CTS, changing paths or moving nodes

- External interference, which can be solved by selecting new channels or changing the channel bandwidth

- Propagation conditions, which can be solved relocating nodes, moving to basic transmission modes

- Queuing, reflecting packet drops in different system queues, which can be solved limiting application demand

5. Field validation and conclusions

During the final project demonstrations conducted at the Almada Camp of the Portuguese Navy and the Roi Albert Camp of the Belgium Army, the ICARUS COM system and associated tools have proven to offer significant value for mission commanders along different mission phases, as illustrated on **Figures 8–10**. First, as a powerful deployment planning tool

Figure 8. ICARUS COM tools communicating with aerial robotic systems (acting as communication relays). (Source: ICARUS).

Figure 9. ICARUS COM tools communicating to rescue workers operating inside a rubble field. (Source: ICARUS).

and second, as a network management and optimisation tool able to seamlessly connect all robots' telemetry and tele-control capabilities to the ICARUS C2I stations, mitigating eventual coverage and throughput shortcomings arising during operations.

The ICARUS communication system makes use of HW/SW mass-market technologies thoroughly engineered for professional performance exploiting unlicensed spectrum in UHF, 2.4 and 5 GHz bands. The "unlicensed spectrum" approach has provided acceptable performance during the set of trials executed during the project life under limited interference conditions. Nevertheless, in real-life safety-critical SAR operations, it is highly desirable having guaranteed access to radio spectrum with proper EIRP limits to ensure required throughput and operation in long ranges or harsh propagation scenarios such as rubble or indoor [28–31]. The ICARUS communication system includes by-design specific provisions to ease integration of new datalink technologies and extend operation to new frequency bands, by adapting the cognitive radio functions to implement any required spectrum access rules. Existing 802.11 COTS professional transceivers that can be tuned to operate in any band up to 6 GHz will allow to

Figure 10. ICARUS COM tools installed on a small unmanned ground vehicle. (Source: ICARUS).

readily reuse all of the COMMW/COMCON 802.11 capabilities in low-frequency spectrum particularly suitable and eventually protected for public protection and disaster relief (PPDR) applications. In the migration phase towards commercialisation, the team is also working on the integration of LTE services; either commercial (if available on crisis location) or PPDR-specific (e.g. operating in the 700 MHz) to be used as a complementary incident-spot capacity as an interconnection means between distant incident-spots. While low-layer LTE functions would be out of control of ICARUS COM reducing optimisation possibilities, the framework is already able to evaluate in real time the throughput and latency offered by external networks, which would be used to manage the available capacity as a whole.

Author details

José Manuel Sanchez[1]*, José Cordero[1], Hafeez M. Chaudhary[2], Bart Sheers[2] and Yudani Riobó[3]

*Address all correspondence to: jose.sanchez@integrasys-sa.com

1 Integrasys SA, Calle Las Rozas de Madrid, Spain

2 Royal Military Academy, Belgium

3 Quobis Networks SL, O Porriño, Spain

References

[1] Stopforth R. van de Groenendaal H. "Search and rescue robot-lessons from 9/11", engineer IT, electronics, computer, information & communication technology in engineering. SAIEE Journal. Jan. 2010

[2] Baldini G, Karanasios S, Allen D, Vergari F. Survey of wireless communication technologies for public safety, IEEE Communications. Surveys & Tutorials. 2nd Quarter, 2014;**16**(2):619-641

[3] Kumbhar A, Koohifar F, Guvenc I, Mueller B. A survey on legacy and emerging technologies for public safety communications. IEEE Communications. Surveys & Tutorials. 1st Quarter, 2017;**19**(1):97-124

[4] Malone BL. Wireless search and rescue: Concepts for improved capabilities. Bell Labs Technical Journal. Apr. 2004;**9**(2):37-49

[5] Fragkiadakis AG, Askoxylakis IG, Tragos EZ, Verikoukis CV. Ubiquitous robust communications for emergency response using multi-operator heterogeneous networks.

EURASIP Journal on Wireless Communications and Networking. 2011;**13:** 1-16. DOI:10.1186/ 1687-1499-2011-13)

[6] Fujiwara T, Watanabe T. An ad hoc networking scheme in hybrid networks for emergency communications. Elsevier Journal on Ad Hoc Networks. Sep. 2005;**3**(5): 607-620

[7] George SB, Whou W, Chenji H, Won Y, Lee O, Pazarlogou A, Stoleru R, Barooah P. DistressNet: A wireless ad hoc and sensor network architecture for situation management in disaster response. IEEE Communications. Magazine. Mar. 2010;**48**(3):128-136

[8] Nelson CB, Steckler BD, Stamberger JA. The evolution of hastily formed networks for disaster response: technologies, case studies, and future trends. In: IEEE Global Humanitarian Technology Conference (GHTC), Seattle, USA. Nov. 2011

[9] Felice MD, Trotta A, Bedogni L, Chowdhury KR, Bononi L. Self-organizing aerial mesh networks for emergency communication. In: IEEE 25th International Symposium on Personal, Indoor, and Mobile Radio Communication (PIMRC), Washington DC, USA. Sep. 2014

[10] Reina DG, Askalani M, Toral SL, Barrero F, Asimakopoulo E, Bessis N. A survey on multi-hop ad hoc networks for disaster response scenarios. International Journal of Distributed Sensor Networks. Jan. 2015;**2015**

[11] Salamanca MDP, Camargo J. A survey on IEEE 802.11-based MANETs and DTNs for survivor communication in disaster scenarios. In: IEEE Global Humanitarian Technology Conference (GHTC), Seattle, USA. Oct. 2016

[12] Minh QT, Nguyen K, Borcea C, Yamada S. On-the-fly establishment of multihop wireless access networks for disaster recovery. IEEE Communications Magazine. Oct. 2014;**52**(10):60-66

[13] Aloi G, Bedogni L, Bononi L, Briante O, Di Felice M, Loscrì V, Pace P, Panzieri F, Ruggeri G, Trotta A. STEM-NET: How to deploy a self-organizing network of mobile end-user devices for emergency communication, Elsevier Journal on Computer Communications. Apr. 2015;**60**:12-27

[14] Ray SK, Sinha R, Ray SK. A smartphone-based post-disaster management mechanism using WiFi tethering. In: IEEE 10th Conference on Industrial Electronics and Applications (ICIEA), Auckland, New Zealand. June 2015

[15] Koumidis K, Kolios P, Panayiotou C, Ellinas G. ProximAid: Proximal adhoc networking to aid emergency response. In: International Conference on Information and Communications. Technologies for Disaster Management (ICT-DM), Rennes, France. Dec. 2015

[16] Bai Y, Du W, Ma Z, Shen C, Whou Y, Chen B. Emergency communication system by heterogeneous wireless networking. In: IEEE International Conference on Wireless Communications. Networking and Information Security (WCNIS), Beijing, China. June 2010

[17] Dalmasso I, Galletti I, Giuliano R, Mazzenga F. WiMAX networks for emergency management based on UAVs. In: IEEE 1st AESS European Conference on Satellite Telecommunication (ESTEL), Rome, Italy. Oct. 2012

[18] Morgenthaler S, Braun T, Zhao Z, Staub T, Andwander M. UAVNet: A mobile wireless mesh network using unmanned aerial vehicles. IEEE Globecom Workshops (GC Wkshps), Anaheim, California. Dec 2012

[19] Flushing EF, Gambardella LM, Di Caro GA. On using mobile robotic relays for adaptive communication in search and rescue missions. In: IEEE International Symposium on Safety, Security, and Rescue Robotics (SSRR), Lausanne, Switzerland. Oct. 2016

[20] Singh A, Adams R, Dookie I, Kissoon S. A simulation tool for examining the effect of communications on disaster response in the oil and gas industry. IEEE Transactions on Systems, Man, and Cybernetics, Aug. 2016;**46**(8):1036-1046

[21] Erdelj M, Natalizio E, Chowdhury KR, Akyildiz IF. Help from the sky: Leveraging UAVs for disaster management. IEEE Pervasive Computing. Jan-Mar. 2017;**16**(1):24-32

[22] Gonzales D, Harting S. Designing unmanned systems with greater autonomy, RAND Corporation. 2014

[23] DMR Association. Benefits and features of DMR. White Paper. Available from: http://dmrassociation.org. Retrieved Dec. 2016

[24] IEEE 802.11™-2012, Standard for information technology-telecommunications and information exchange between systems local and metropolitan area networks-specific requirements Part 11: Wireless LAN medium access control (MAC) and physical layer (PHY) specifications. Available from: https://standards.ieee.org

[25] Chaudhary MH, Scheers B. Dynamic channel and transmit-power adaptation of WiFi network in search and rescue operations. In: International Conference on Military Communications and Information Systems (ICMCIS), Brussels, Belgium. May 2016

[26] Ghafoor S, Sutton PD, Sreenan CJ, Brown KN. Cognitive radio for disaster response networks: Survey, potential, and challenges. IEEE Wireless Communications. Oct. 2014;**21**(5):70-80

[27] Chaudhary MH, Scheers B. DMR implementation on SDR platform for control of unmanned devices and cognitive management of WiFi network in search and rescue missions. In: NATO IST/RSY Symposium on Cognitive Radio & Future Networks, The Hague, The Netherlands. May 2015

[28] Holloway CL, Koepke G, Camell D, Young WF, Ramley KA. Radio propagation measurements during a building collapse: Applications for first responders. In: International Symposium on Advanced Radio Technology (ISART), Boulder, USA. Mar. 2005

[29] Oestges C. Radio channel models for search-and-rescue missions into collapsed structures. In: URSI International Symposium on Electromagnetic Theory (EMTS), Hiroshima, Japan. May 2013

[30] Anusas-amornkul T. A victim and rescuer communication model in collapsed buildings/ structures, In: IEEE International Conference on Parallel and Distributed Systems (ICPADS), Hsinchu, Taiwan. Dec. 2014

[31] Propagation data and prediction methods for the planning of indoor radio communication systems and the radio local area networks in the frequency range 900 MHz to 100 GHz. ITU-R Recommendations, Geneva. 2001

Command and Control Systems for Search and Rescue Robots

Shashank Govindaraj, Pierre Letier,
Keshav Chintamani, Jeremi Gancet,
Mario Nunez Jimenez, Miguel Ángel Esbrí,
Pawel Musialik, Janusz Bedkowski, Irune Badiola,
Ricardo Gonçalves, António Coelho, Daniel Serrano,
Massimo Tosa, Thomas Pfister and
Jose Manuel Sanchez

Additional information is available at the end of the chapter

Abstract

The novel application of unmanned systems in the domain of humanitarian Search and Rescue (SAR) operations has created a need to develop specific multi-Robot Command and Control (RC2) systems. This societal application of robotics requires human-robot interfaces for controlling a large fleet of heterogeneous robots deployed in multiple domains of operation (ground, aerial and marine). This chapter provides an overview of the Command, Control and Intelligence (C2I) system developed within the scope of Integrated Components for Assisted Rescue and Unmanned Search operations (ICARUS). The life cycle of the system begins with a description of use cases and the deployment scenarios in collaboration with SAR teams as end-users. This is followed by an illustration of the system design and architecture, core technologies used in implementing the C2I, iterative integration phases with field deployments for evaluating and improving the system. The main subcomponents consist of a central Mission Planning and Coordination System (MPCS), field Robot Command and Control (RC2) subsystems with a portable force-feedback exoskeleton interface for robot arm tele-manipulation and field mobile devices. The distribution of these C2I subsystems with their communication links for unmanned SAR operations is described in detail. Field demonstrations of the C2I system with SAR personnel assisted by unmanned systems provide an outlook for implementing such systems into mainstream SAR operations in the future.

Keywords: command and control, human machine interfacing

1. Introduction

This chapter describes the concepts and features behind the command, control and intelligence (C2I) system developed in the ICARUS project, which aims at improving crisis management with the use of unmanned search and rescue (SAR) robotic appliances embedded and integrated into existing infrastructures. A beneficial C2I system should assist the search and rescue process by enhancing first responder situational awareness, decision-making and crisis handling by designing intuitive user interfaces that convey detailed and extensive information about the crisis and its evolution.

The different components of C2I, their architectural and functional aspects are described along with the robot platform used for development and field testing in **Figure 1**. This section also provides an elicitation and analysis of the ICARUS C2I system requirements and the overall system and subsystem components' architecture (hardware and software), along with the interfaces and data shared between these components. The objective is to provide a static and dynamic view of the structure and hierarchy within the components of this system.

There have been recent efforts [1, 2, 3] where C2I robots have been deployed for SAR, but the focus was mainly on human-robot cooperation, and there is no holistic approach to enable control of heterogeneous robotic assets. The requirement for customized robots and their control centres, equipped to provide a comprehensive common operational picture (COP) for SAR, is being addressed by the ICARUS C2I solutions.

In a disaster struck area, the local emergency management authority (LEMA) is responsible for the overall command, coordination and management of the response operation. The C2I system will provide extensive interfaces to incorporate unmanned systems, for augmenting the capabilities of SAR operation planning and execution. The seamless integration of human SAR teams with unmanned platforms is an integral feature of the C2I system [4].

The C2I system of ICARUS [5] consists of a central mission planning and coordination system (MPCS), field portable robot command and control (RC2) subsystems, a portable force-feedback exoskeleton interface for robot arm tele-manipulation and field mobile devices. The deployment of C2I subsystems with their communication links for unmanned SAR operations is shown in **Figure 2**.

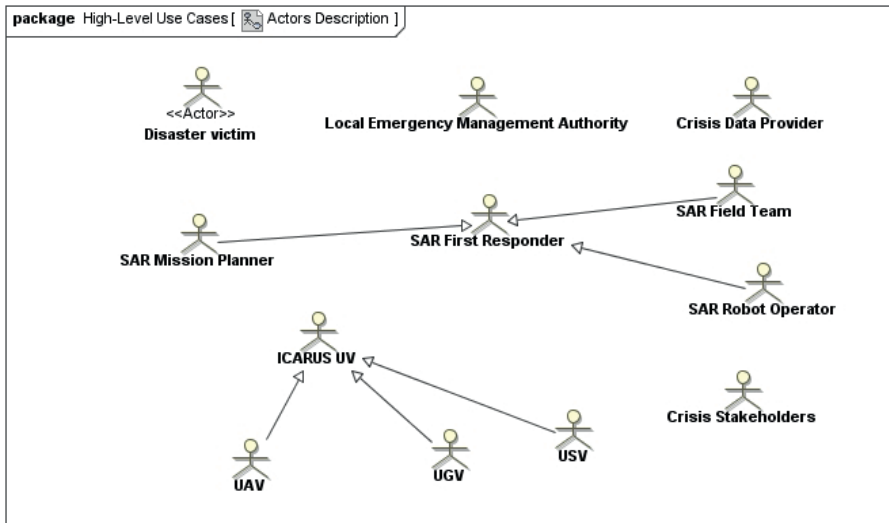**Figure 1.** C2I deployment and communication framework (source: ICARUS).

Figure 2. Main actors involved in the C2I high-level use cases (source: ICARUS).

2. Approach to designing the C2I

2.1. State of the art

Abstract mission planning and supervisory control is essential for deploying multiple unmanned systems for reconnaissance and mapping tasks, in large and open environments for extended durations. Commercial ground control stations are available for controlling and planning missions for single unmanned aerial vehicles (UAVs) such as Portable Ground Control Station [6] and OpenPilot GCS [7]. The availability of multi-UAV [8] base control stations is not widespread, but limited to a few such as the QGroundControl [9]. Apart from allowing users to plan UAV missions, these utilities are primarily designed for UAV development, debugging and testing. Supervisory interfaces for robot systems have been designed and developed, for instance, the DexROV [10] control centre, to perform offline system training and online task monitoring for remote ROV operations. These interfaces still require humans constantly in the loop for performing low-level tasks and for coordinating tasks between unmanned systems. Deployment of Unmanned Ground Vehicles (UGV), Unmanned Aerial Vehicles (UAV) and Unmanned Surface Vehicles (USV) autonomously for long-endurance operations requires an approach as described in the Multimodal User Supervised Interface and Intelligent Control (MUSIIC) project [11]. An ecological interface [12] design analysis [13] management operator centric needs will need to be performed to evaluate how the human cognitive system imposes constraints on the processing of information from multiple unmanned assets [14, 15]. Identifying the three levels of cognitive control—skill-based,

rule-based and knowledge-based—is important for ensuring effectiveness of the supervisory control system for managing the unmanned fleet [16]. Displays for integrating information from different frames of reference, exocentric and egocentric, present potential human performance issues which need to be carefully evaluated [17]. The supervisory control centre will be used only for high-level global mission planning and monitoring. The central command and control base station will be deployed near the port, capable of planning missions for UAVs and USVs to execute their tasks cooperatively. The graphical interface will be designed based on ecological design concepts [15] to improve situational awareness.

2.2. End-user involvement

Inputs and consideration of end-user requirements for the C2I system design are critical as it is the principle interface between them and the unmanned platforms in SAR scenarios. The ICARUS C2I [18] is a complex system providing the end-users with multiple user interfaces at various operation levels. For example, the MPCS is aimed at mission managers and mission planners; the RC2 is aimed at robot operators, and the mobile application is for rescue workers. Work in the field of robotic control (user) interfaces has, for a long period, remained a research topic. Most user interfaces in use today are designed for specific end-users (fire fighters, soldiers, etc.), robotic platforms [unmanned ground vehicles (UGVs), USVs and UAVs] and applications (e.g. Explosive Ordnance Disposal EOD, reconnaissance and surveillance). In ICARUS, the challenge is to develop a unified system that enables control of heterogeneous robotics platforms.

For this complex system to work well with end-users, a user-centred design approach has been adopted. Contact was established with end-users to understand SAR processes and methods early in the project. Only after meetings with end-users and reviews of operational scenarios in the INSARAG guidelines was the concept for the ICARUS C2I proposed. The system requirements have been derived from the user requirements collected in the initial phases of the project. The system concept has been reviewed by B-FAST members with the general approach. However, it must be noted that the bespoke nature of the C2I, the unavailability of reference implementations and low user experience with robotic platforms make it difficult for end-users to provide usable feedback before early system prototypes are available. The approach taken was to invite end-users to review early prototypes and gather their feedback by initiating dialogs with B-FAST and setting up user-review meetings frequently.

2.3. High-level and detailed use cases

The high-level use cases of the ICARUS C2I system describe the main interactions of the system with the various actors (SAR users and other systems). The objective of the high-level use cases is to ensure that the C2I design concept adequately covers the main needs for Urban Search and Rescue (USAR) and Maritime Search and Rescue (MSAR) operations. It must be noted that the high-level use cases provide the reader with a broad view of the interactions of the different actors with the C2I. The main actors and their interrelationships are provided in **Figure 2**. The following actors are envisaged as the main users of the C2I system:

- Disaster victim

- Local emergency management authority (LEMA)

- Crisis data provider

- SAR first responder

- SAR mission planner

- SAR robot operator

- SAR field team

- Crisis stakeholders

- ICARUS unmanned vehicle (UV): UGV, UAV and USV

These use cases are developed under three main packages which have been identified based on the proposed USAR scenarios:

2.3.1. Mission planning and control

This package covers the use cases (**Figure 3**) of the C2I system in the context of mission planning. Mission planning will be the first task undertaken after setup of the hardware which includes and is not limited to disaster data analysis, area reduction, resource assessment and assignment, monitoring and coordinating actors and systems in the field, communications with stakeholders and revising and updating mission plans.

2.3.2. Robot command and control

The main robot command and control interactions with the actors and the C2I system are described in **Figure 4**. As a high-level use case, this includes the control of all ICARUS robotic
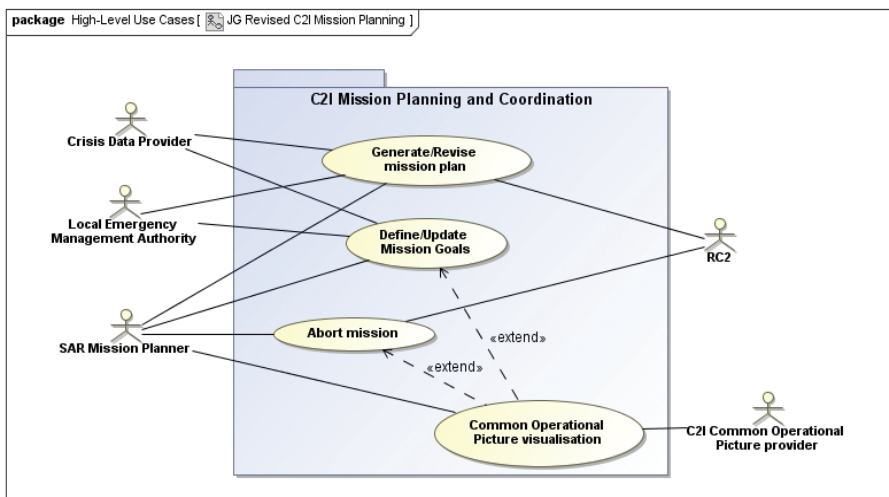

Figure 3. C2I high-level use cases for mission planning and coordination (source: ICARUS).
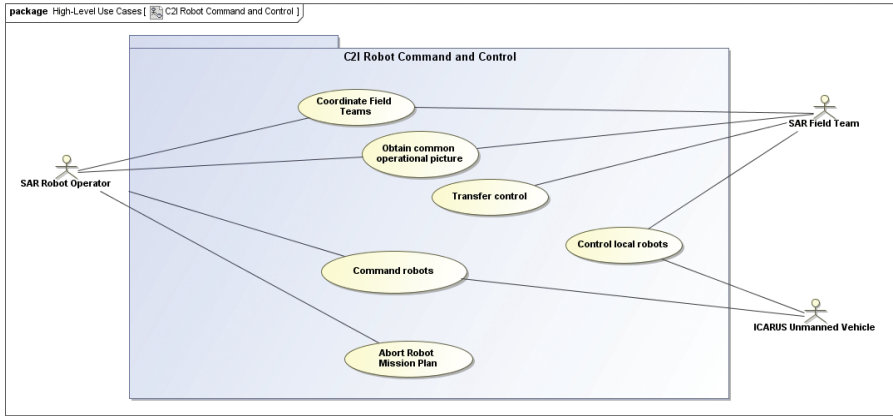
Figure 4. C2I high-level use cases for robot command and control (source: ICARUS).

systems. The following use-case packages have been identified to categorically group the interactions of the robot operator with the RC2 system:

- **Robot mission execution**: Tasks performed before and during the period one or more robots are deployed in a disaster zone.

- **UAV command and control**: These use cases describe the various interactions foreseen for UAV guidance, navigation and control.

- **UGV command and control**: The various interactions foreseen when the robot operator uses the UGVs for search and rescue operations.

- **USV command and control**: The use cases describe the interactions of the robot operator with the different unmanned surface vehicles.

- **Heterogeneous command and control**: The use cases specify the interactions of the robot operator under conditions where cooperative behaviour between pairs of robots is foreseen.

2.3.3. Mobile interface for SAR responders

Figure 5 describes the principal lines of interactions for exchanging data between the C2I and field deployed actors to receive an updated common operational picture (COP) and to push updates to the C2I from field operations.

2.4. Subsystem analysis

The C2I system will provide a variety of functions for SAR teams under the global objective of identifying disaster victims in a fast and efficient manner. Based on the high-level use-case analysis, the requirements can be classified and grouped into six major groups:
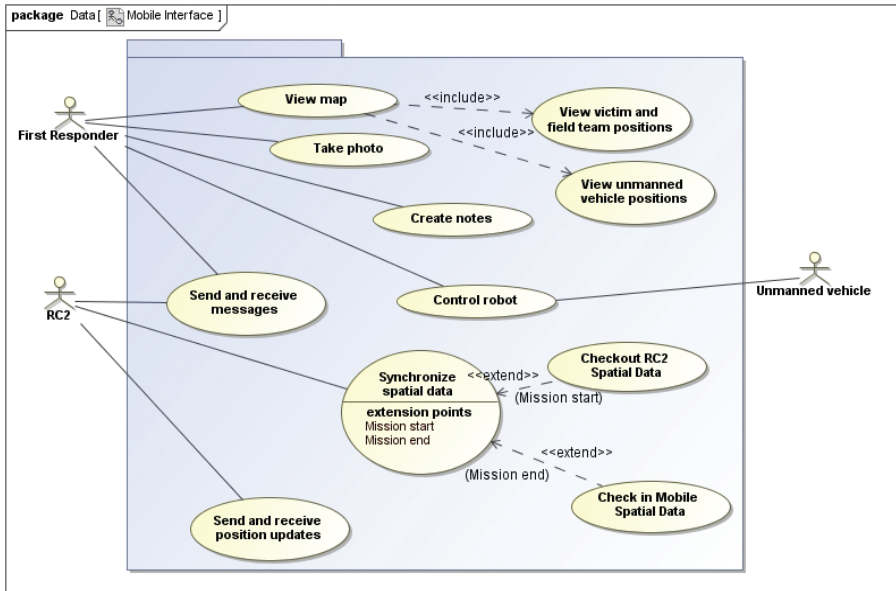
Figure 5. Mobile interface for first responders' main use cases (source: ICARUS).

1. Mission planning and coordination tools and subsystems.

2. Command and control subsystems for unmanned vehicle control. This includes a force-feedback system for control of the robot arms mounted on the UGVs.

3. A mobile application to enable communications between the above systems and first responders working at the intervention site.

The main functionality provided by each of the above systems is described in the following sections.

2.4.1. Mission planning and coordination system (MPCS)

The mission planning and coordination requirements for the C2I system illustrate the need for the availability of tools to help SAR mission planners to organize and deploy SAR human and robot teams in a disaster zone. Extending the requirements, this means that the C2I system must include a subsystem that allows SAR mission planners to create mission plans, monitor missions and make decisions to update or abort missions [19]. This subsystem is titled as the mission planning and coordination subsystem (MPCS). The system provides the SAR mission planner with the ability to allocate SAR resources based on an analysis of crisis data. SAR resources could be allocated to specific crisis 'sectors' that are designated as critical by the SAR mission planner with the support of the MPC tools. During a mission, the MPCS allows the SAR mission planner to monitor the progress of the field and robotic

teams, simultaneously enabling the SAR mission planner to reallocate resources or add more resources to one or more sectors. During mission progress, the SAR mission planner would be able to communicate with the field teams. The MPCS is based on human in the loop intelligent planning systems to automate several high-workload tasks [20] that are usually required to be performed manually by the SAR mission planner.

2.4.2. Robot command and control (RC2)

The RC2 subsystem's primary aim is to provide the robot operator with the interfaces needed for safe monitoring and control of the heterogeneous set of ICARUS robots. For robot command and control tasks, the RC2 subsystem encompasses all the functionality that is needed for the operator to monitor and coordinate the robot operations in the disaster zone. The RC2 will also serve as the server for the mobile interfaces, routing and updating the field teams through the mobile devices. In addition, specific functionality to allow the robot operator to communicate with disaster victims must also be considered in the design process [21].

The robot operator is the main actor who is envisioned to use the RC2 system. He will command and control the various unmanned platforms in ICARUS. Mission level directives and mission plans will be provided to the robot operator by the SAR mission planner who operates the MPC subsystem at the on-site operations coordination centre (OSOCC). For manual or semi-manual tele-operation of the robotic platforms, the robot operator will use input interfaces as tactile devices, joysticks or force-feedback exoskeleton arms in the case of the control of a slave robotic arm mounted on top of the mobile platforms. With its anthropomorphic configuration, this solution offers a very intuitive manner to control the slave robot arm. It enables also precise force interaction with the environment with the purpose to reduce the risks of accidents and improve operation efficiency.

2.4.3. Mobile application for first responders

End-users have expressed their interest in a mobile application that allows them to carry a digital map of the disaster sector given that most of them have a smartphone or similar device that allows viewing of such data. The mobile interface has been developed that caters to this need from the end-users however with additional functionality. The mobile application will provide a map viewer through which the user can view, for example, the activity of other field teams, identified victim locations and the positions of the various robots in the vicinity.

Other optional data layers could be considered such as weather overlays and updated satellite imaging of the disaster area. In addition, the mobile application will allow the user to receive updates from the robot operator about the progress of an ongoing mission. The system also allows the user to send messages to the robot operator which includes field observations to improve the situational awareness of the robot operator.

2.4.4. Exoskeleton with force feedback

The arm force-feedback exoskeleton is an advanced Human Machine Interface (HMI) allowing the operator to intuitively control slave robotic arms such as the one that will be

mounted on the large UGV platform. The main purpose of the exoskeleton during standard operation is to:

- Measure position of operator's arm to send this as a command to move the slave robotic arm.

- Produce force feedback on the operator as a rendering of the forces exerted on the slave device, as guiding feature for advanced operations or for safety purposes (limits of workspace).

The exoskeleton subsystem is composed of several components:

- The exoskeleton device itself, including sensor, actuators and low-level electronics.

- The exoskeleton controller, responsible for the communications with the RC2 and the computation of the high-rate haptic loop.

- The powering unit to deliver the required power to the exoskeleton.

2.5. Deployment scenarios

It is a common knowledge that there is no easy way to generalize a natural disaster and its effects. Several parameters affect SAR work including coverage area, disaster source, terrain characteristics, etc. Following the INSARAG guidelines, the general procedure followed by international teams is to arrive at the affected country and set up an on-site operations coordination centre (OSOCC) close to the disaster zone. The OSOCC then coordinates and controls the SAR activities for a given disaster zone. In the case where the disaster area is large, sub-OSOCCs are formed at designated disaster sectors.

Given this organizational structure in SAR tasks, it is important to design ICARUS C2I components so that a similar structure can be implemented in the coordination, command and control of robotic systems during a crisis [22]. In this regard, two scenarios of C2I deployment are foreseen with the different subsystems proposed in the previous section which are in line with standard SAR operating procedures. Another determining criteria for these scenarios are due to the constraints posed for communication between the various robotic and C2I systems during a SAR mission. The two envisioned scenarios that the C2I system should support are provided below.

2.5.1. Centralized command and control

In the first case, it is assumed that the OSOCC is located within 1 km of all disaster zones. In this situation, the SAR mission planner using the MPCS and the robot operator using the RC2 and the exoskeleton will be located at the OSOCC with the field teams and robots performing SAR operations in nearby designated disaster sectors. The main operational constraints are (1) sufficient data bandwidth to permit monitoring and control of the robots, (2) a high-frequency channel for force feedback between the robot arms and the exoskeleton and (3) data transfer between the RC2 and the mobile devices. It must be kept in mind that in this scenario, the RC2 will be used primarily for non-line-of-sight robot operations. **Figure 6** provides a schematic diagram of this scenario.
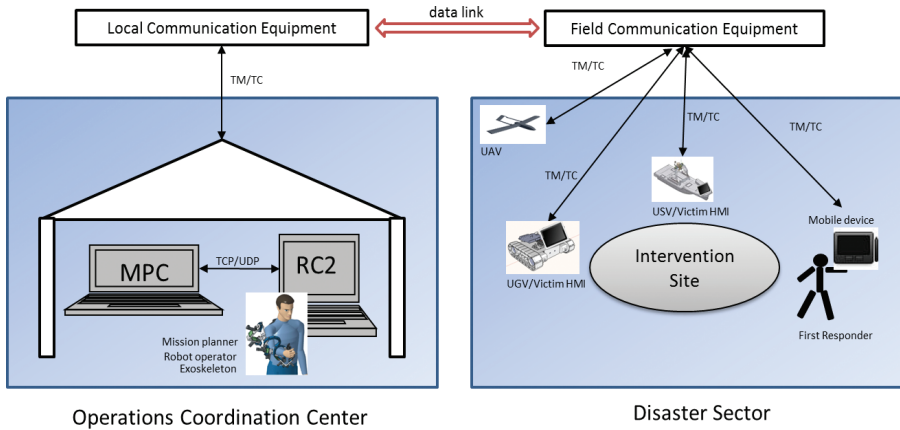
Figure 6. Deployment scenario of ICARUS C2I for SAR operations close to the OSOCC (source: ICARUS).

The SAR mission planner observes the progress of the mission using the MPCS and updates the mission plans. The mission plans are provided to the RC2 system, which the robot operator uses to issue commands and monitor the progress of the robots. In each disaster zone, one or more first responders can carry a mobile device which executes the mobile application. The mobile devices will provide mission-specific data to the robot operator who then uses the information to coordinate the robots. Frequent information exchange is foreseen between the robot operator and the SAR mission planner.

2.5.2. Distributed command and control

The aim of this scenario is to provide a C2I system that can cater to the needs of a range of disaster situations, thus providing flexibility and extensibility. When a disaster scenario covers a large area or when the disaster sectors are located at distances greater than 3 Km, it might not be feasible for the robot operator to be located at the OSOCC. The reason for this is that the latency in communication will affect the ability to perform time-critical operations with the robots.

In the distributed command and control scenario, the MPCS is located at the OSOCC and is used by the SAR mission planner to generate a mission plan. The RC2 receives at predetermined frequencies mission updates from the MPCS. The robot operator then executes the mission plan by deploying the ICARUS robots at the intervention site. In this distributed concept, multiple RC2 systems can be deployed, each servicing a unique disaster zone. In each disaster zone, one or more first responders can carry a mobile device which executes the mobile application. The scenario is depicted in **Figure 7**.

The distributed command and control scenario uses a hierarchical approach for data exchange. The MPCS coordinates and serves as the data server for all RC2 systems, and similarly the RC2 serves as the data coordinator for the mobile devices and the robot-victim HMI, along with hosting the robot platform-specific data.
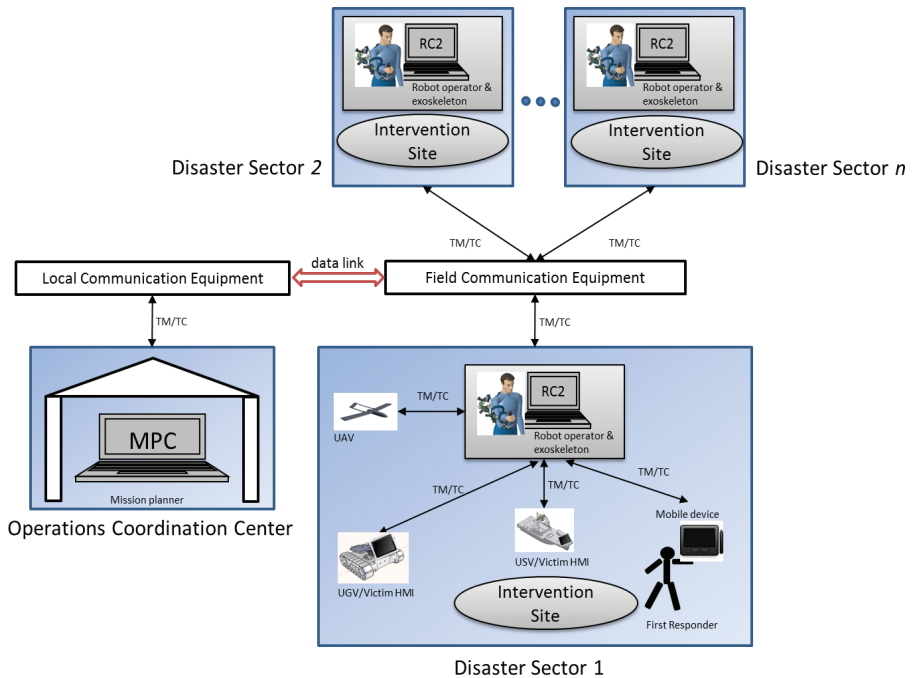
Figure 7. Distributed scenario for SAR operations performed at different sectors (source: ICARUS).

3. C2I system architecture

3.1. Deployment architecture

The main subsystems of the C2I were identified earlier in Section 2.3 where preliminary descriptions of the features of these systems were provided. **Figure 8** presents the deployment architecture of the interconnected C2I subsystems. The MPCS is a stand-alone software application that will run on a Windows or Linux workstation located at the OSOCC. It will use Ethernet (IEEE 802.3) or Wireless LAN (IEEE 802.11b/g/n) to share data between the various RC2 systems deployed in the field. The SAR mission planner located at the OSOCC updates the latest crisis data on the MPCS and generates a mission plan for a given sector or sectors. Mission plans and crisis data are distributed to the various RC2 systems via a distributed geospatial information systems (GISs). The MPCS will also have a continuously open link with one or more RC2 systems to send and receive data.

The RC2 application will be executed on a ruggedized laptop designed for outdoor use, keeping in line with the user requirements for non-LOS and LOS (Line of Sight) robot tele-control. One of its main purposes is to synchronize mission plans and crisis data relevant to the sector
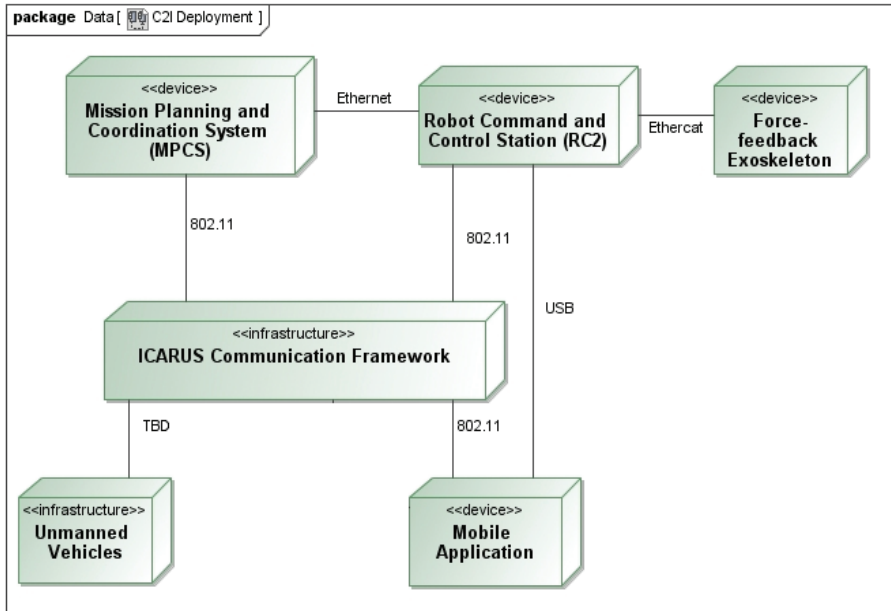
Figure 8. Deployment architecture of C2I subsystems (source: ICARUS).

it is designated for, with the MPCS. It is foreseen that the RC2 could be located at the OSOCC, alongside the MPCS or in a remote mode, where it links to the MPCS via the ICARUS communication framework. The RC2 pushes knowledge of the sector's mission progress to the MPCS. The RC2 hosts data critical for the operation of the following hardware: (1) ICARUS robots, (2) the exoskeleton and (3) mobile devices in the field. One of the primary aims of the RC2 is to provide robot operator with intuitive tools to command and control multiple, heterogeneous robots. In addition, it allows first responders with mobile devices to receive the latest mission updates and sectors maps.

Using a mobile device, first responders can push and pull messages, photos and position information over the network to the RC2. All mobile devices will connect via a Transmission Control Protocol (TCP) link (wireless) to the RC2 system. The exoskeleton interfaces with the RC2 using an EtherCAT interface, providing high-fidelity haptic rendering and manipulation capabilities for robotic arm control. The RC2 provides the visual interfaces for visualization of robotic arm movement. In the C2I architecture, robot manipulation, control and sensor data handling are restricted to the RC2.

3.2. Functional software components

The MPCS and RC2 are designed to have a distributed architecture where different components (processes) have control and data interfaces. The robot operating system (ROS) middleware

has been chosen to implement the C2I components. The motivations behind the adoption of a distributed framework like ROS are the following:

- To maximize the reusability of available robot sensor visualizations, sensor fusion and control algorithms.

- To adopt a standard framework used extensively on robotic platforms.

- This approach is coherent for rapid integration of the C2I with diverse robotic platforms in different deployment scenarios and provides a flexible approach in comparison with contemporary solutions. Existing robot command and control centres are either coupled to a specific robot platform or fixed to a specific SAR deployment scenario.

- Different modules can be developed separately by partners adhering to the ROS architecture and integrated easily within the C2I system.

- ROS defines standard message types for commonly used robot sensor data such as images, inertial measurements, GPS, odometry, etc. for communicating between nodes. Thus, separate data structures need not be explicitly defined for integrating different components.

The MPCS and RC2 user interfaces enable the SAR mission controller to maintain a common operational picture (COP) and manage the execution, coordination and planning of the SAR operation [23]. In **Figure 9**, different ROS components of the RC2 system have been illustrated at a high level using the ROS framework. A high-level description of each component will be given in the following subsections.
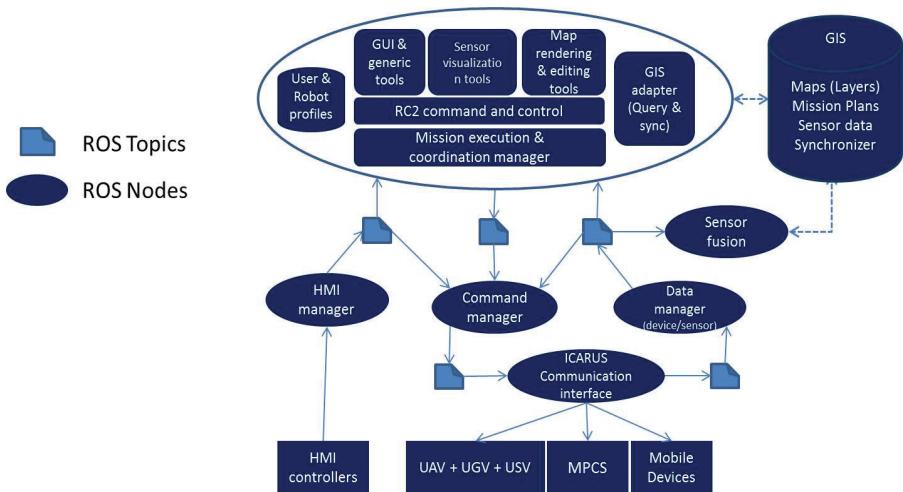

Figure 9. RC2 subsystem components (source: ICARUS).

3.2.1. Mission planning and coordination system (MPCS)

The MPCS gathers functionalities allowing the specification and management of missions during their execution at the OSOCC level. **Figure 10** describes components supporting the assembly analysis of data collected from the mission sections [a.k.a. common operational picture (COP)], the visualization/rendering of these data by users, the specification of mission objectives relying on these data and the planning of mission tasks based on specified objectives and high-level monitoring of mission execution [24]. The MPCS is primarily connected to the SAR first responders—essentially embodied as RC2s. Some of the major components are described below:

Mission goals specification tool: This component gathers functions required to specify mission goals. It gathers the main components of the mission goals specification interface, offering dedicated tools for goals definition, a mission specification database where the mission goals are stored and a watchdog monitoring the evolution of the mission execution. Live mission data material, under all available forms: images, various measurements, symbolic and abstract representations, streaming (visual and/or aural), etc.

Watchdog: The watchdog monitors the evolution of the mission execution, possible issues in plan being executed and needs for, e.g. constraints relaxation. The watchdog provides notification of potential issues to the users, so that actions can be taken to update the mission goals accordingly.

Mission goals specification interface: Provides the primitives for ICARUS mission goals identification, such as inspection of a zone, surveillance of a zone, request of perception with
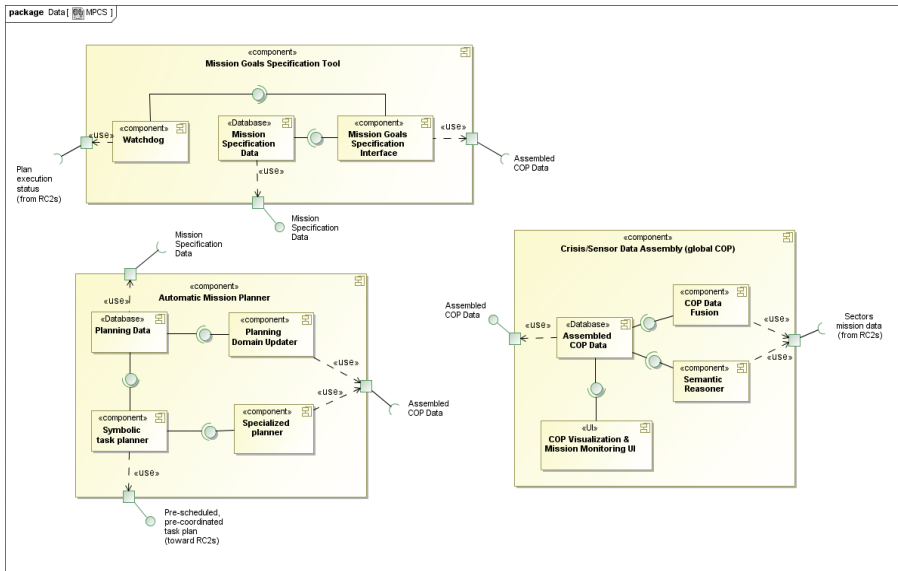

Figure 10. MPCS subcomponents and their interfaces (source: ICARUS).

certain modalities (e.g. panoramic view) from a given location, etc. Constraints can in addition be specified, such as time extent, robotic platform preferences, human team composition or preferences, etc.

Mission specification data: GIS database storing mission specification data as provided from the mission goals specification interface.

Automatic mission planner: This is a central component that is capable to turn the high-level mission objectives into RC2-level executable task details, which are both pre-coordinated and prescheduled. This means that resulting data are ready for execution while having flexibility in the plan expression (time flexibility, through timelines). It consists essentially of a planning problem builder subcomponent, a symbolic task planner engine and a set of specialized planners supporting the main symbolic planner [25].

Planning domain updater: The planning domain updater's main duty is to maintain the symbolic representation of the 'word', i.e. the environment and actors, while events and changes occur.

Planning data: GIS database storing the expression of planning domain and problems, accordingly providing material to the symbolic task planner as required.

Symbolic task planner: The symbolic task planner is a major component of the MPCS. This planning engine takes planning data material as input and generates symbolic task plans in which execution (by robots and/or human team) should allow reaching related mission goals.

Specialized planners: The specialized planners are a set of tools with dedicated functions for computing the cost (and possibly modalities), with a set of robot(s) along with the related agent(s) and environment model, to perform particular tasks, e.g. surveillance, inspection, perception making, navigation to a given location, etc. Algorithms used with the specialized planners should allow near-real-time computation, in order to minimize the time required for generating plans with the symbolic planner.

Crisis/sensor data assembly (Global COP): This deals with the gathering, processing, assembling and providing interfaces for live mission information (as provided by the RC2s)—maintaining a consistent overall picture.

COP data fusion: This component will collate live mission information from the different RC2 systems deployed in the field and store it in the assembled COP database for its later access by the mission specification tool and SAR mission planner. This information also gets displayed in the UI. The COP data fusion processes data related to the mission progress and associated events.

Semantic reasoner: This analyses and generates semantic information/knowledge [26] from the mission information provided by the RC2s. The main source of data is sensor information from the robots and GIS (data stored in database). Reasoner analyses the data and creates semantic model of the environment. The model may be represented in multiple forms: 2D/3D semantic map, enhanced sensor data, enhanced GIS maps, etc. The reasoner will compute steps within a maximum of 10 s.

Assembled COP data: This assembles classical and semantic data into a global COP data source that can be exploited by all other MPCS components as required and that is also used to support user's decision-making (through the user interface). The system will decide which version of semantic information to use: simplified or full.

COP visualization and monitoring UI: Main visualization and monitoring interface for the MPCS. This provides all needed interfaces for the user, as far as mission monitoring is required.

3.2.2. Robot command and control (RC2)

A UML component diagram provided in **Figure 11** describes the RC2 software architecture.
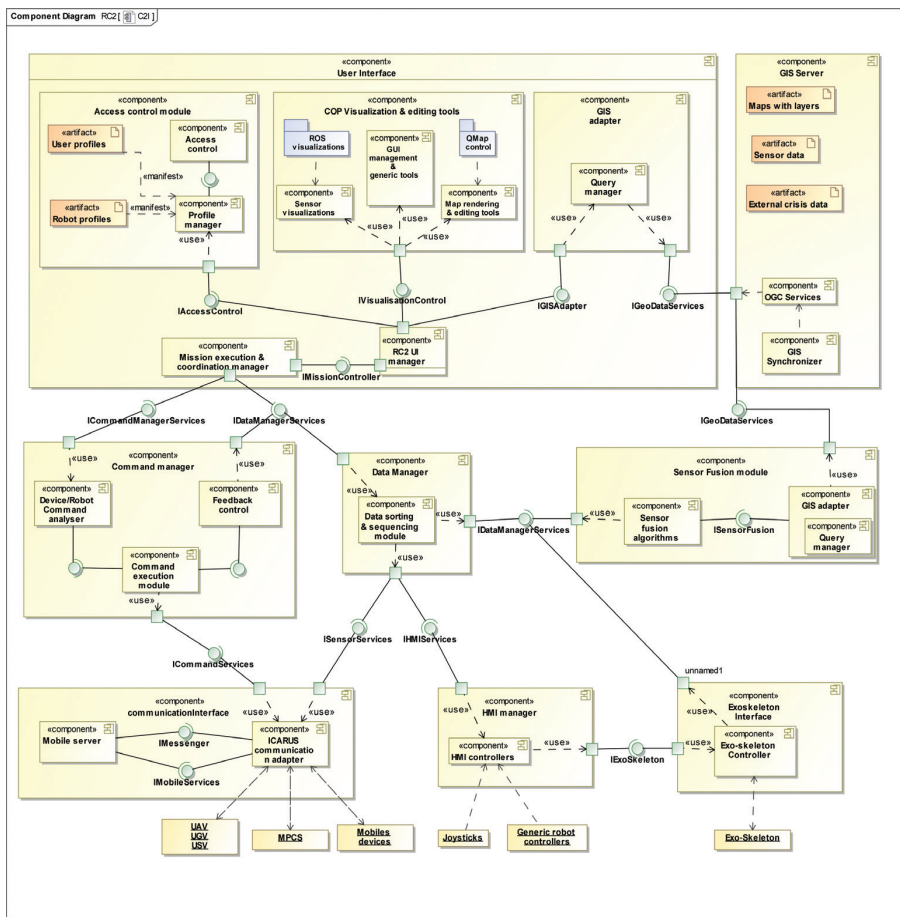

Figure 11. RC2 subcomponents and their interfaces (source: ICARUS).

User profiles: SAR first responders have designated SAR mission planners from LEMA. Authorized SAR mission coordinators are the MPCS and RC2 administrators. An administrator should also have the capability to add new users to access this system. Thus, an access control mechanism is needed to ensure that only authorized users can use this system. This subcomponent of the user interface uses a local encrypted repository to store and retrieve the user profiles primarily consisting of C2I system access control information. A graphical user interface will be provided to (i) login to the C2I, (ii) add or create a new user, (iii) delete an existing user and (iv) modify the access information of an existing user (e.g. change of password).

Access control module: The access control module provides access control functionality in the RC2 system. Its aim is to use a SQLite database to manage user profiles and provide a GUI for users to log in and log out. Although not an explicit user requirement in the project, basic security features will be implemented via this module.

Robot profiles: The C2I system is used to communicate and control heterogeneous robot platforms such as UAVs, UGVs and USVs, with each system having different capabilities (e.g. autonomous, semi-autonomous and tele-operated), sensors and platform-specific concepts. This information is important for planning a mission based on robot capabilities and types of commands that it can execute. Robot profiles will be gathered from all the robotic platforms deployed within the ICARUS framework and stored in a local repository. A generic ROS message schema has been designed (refer to 'Interoperability' section) to dynamically include the features of each robot into the RC2.

Mission execution and coordination manager: This module is specific to the RC2 with a functionality that is a subset of the Global SAR mission coordinator. It has a local view of the SAR mission related to its assigned sector unlike the MPCS, which has a global view of the SAR mission distributed among sectors. It is responsible for triggering the exchange of information between robotic platforms and SAR team members for a coordinated approach to address the mission [23].

GIS adapter: The GIS adapter is responsible for creating queries to the local GIS repository based on requests from the map and robot sensor visualizations. This module receives a set of query parameters, and an appropriate query string will be generated to extract information from the GIS. The GIS provides multiple interfaces for accessing data such as the open geospatial consortium (OGC) standard interface (for maps) and a set of legacy services, to access dynamically generated geo-resources (geo-tagged sensor data and images).

Map rendering and editing tools: A central map widget will be developed to render global base maps using open street maps (OSM) from a local GIS repository. This widget can display aerial maps (captured by unmanned aerial vehicles) overlaid on the base maps. The map will be used to display the locations of unmanned systems and human SAR personnel based on their GPS locations. Tools will be developed for adding waypoints on the map, sectoring areas by drawing polygons, taking geo-tagged notes, tagging images, setting transparencies for different layers and enabling/disabling path tracking for human and unmanned SAR entities [27].

Data manager: The ICARUS communication framework provides a link for receiving data from SAR teams and unmanned platforms. This data is encapsulated in the format defined by the JAUS standard data formats. Message generating modules on deployed ICARUS systems publish geo-tagged sensor data, crisis map updates and other types of data such as voice and images. The data manager at the C2I side is responsible for:

- Decoding or de-serializing sensor data received from robots within the ICARUS communication framework.

- Decoding commands and its associated data, sent between the MPCS and RC2.

- Identifying nodes in the C2I system which can use different types of data.

- Forwarding/channelling de-serialized data across appropriate topics.

This component will provide the main software interface for access to robot sensor data and GIS data. The data manager will provide services for clients to access online as well as offline sensor data. For online sensor data, clients will be able to access RGB (mono and stereo), IR and depth map data available on a specific robot. The sensor manager provides a gateway between crisis data updates received from the MPCS and the geospatial/sensor record database. Live sensor data will be routed to the sensor fusion algorithm component.

Sensor visualization and associated tools: Robot sensor visualizations from the RVIZ-ROS framework are reused and adapted for ICARUS robotic platforms. Existing visualization plugins for 3D point clouds, robot models, grid maps, camera view, etc. will be enhanced with features to improve usability and clarity for the C2I operator. Custom visualization plugins will be developed for robot pose (roll, pitch and yaw), network quality, power status, digital compass, etc. Tools associated with visualizations include 3D image viewpoints, user annotations (points, lines or text), plugin settings, add/remove plugins, etc.

HMI manager: The Human Machine Interface (HMI) manager manages inputs and outputs, from and to HMI devices, respectively. Input devices consist of robot controllers for unmanned systems such as:

- Joysticks

- 3D haptic controllers

- Exoskeleton (joint positions and forces)

- IMU inputs from head-mounted displays (HMDs)

Feedback or outputs from sensors on unmanned can be provided to HMI interfaces capable of rendering them such as:

- Wearable heads-up display (video feeds, robot pose)

- Exoskeleton (haptic force feedback, joint encoder positions)

- Force feedback joysticks

- Calibration of joysticks

The HMI manager in **Figure 12** manages bidirectional data flow between HMI devices and unmanned systems and encodes data depending on the device. For example, control inputs for robots and their peripheral actuators (e.g. robotic arm mounted on a UGV) need to be scaled or interpreted according to the type of end effector. The HMI manager is essentially a ROS node that subscribes to other ROS nodes driving their respective HMI devices. The following diagram illustrates the high-level distribution of the HMI manager with respect to its child nodes.

Platform command manager: This component provides and manages the software interfaces between the robots and the C2I. The platform command manager sequences the commands (scripts, waypoints) through the communication manager to the robots. In its current form, this component is an abstraction for interfaces that receive robot-specific commands. The component handles temporal sequencing of the command data using signals fed forward by the mission execution controller.

Command analyser: The coordinated command generator is a component that will manage cooperative behaviour between pairs of robots such as a UAV and UGV or a UAV and USV. Its purpose is to receive mission-specific coordinated task commands from the user via the command and control UI. It uses instances of the platform command manager to coordinate
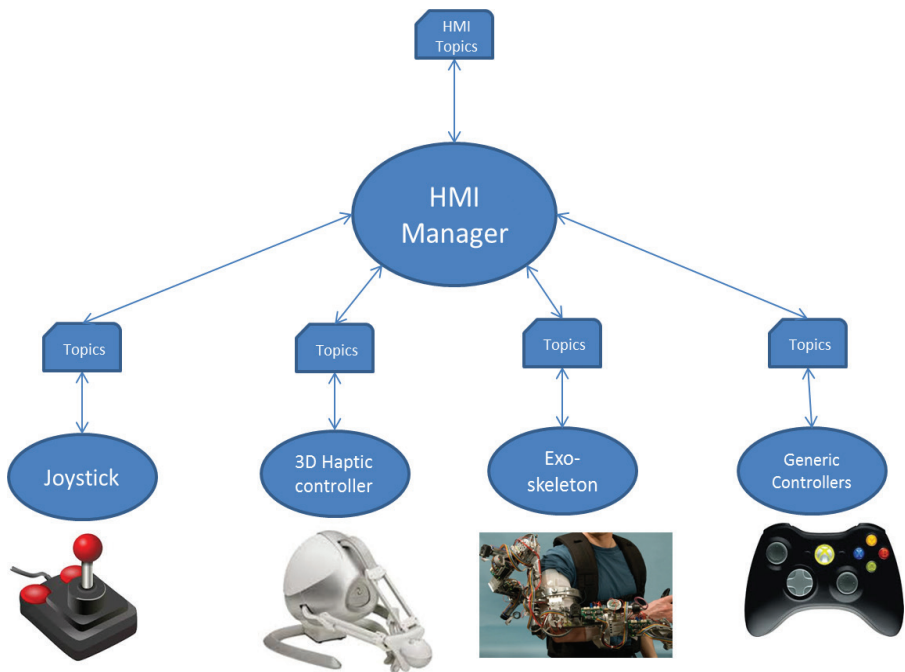

Figure 12. HMI Manger node and its child ROS nodes (source: ICARUS).

command execution between a pair of unmanned platforms. This includes data synchronization between robots.

Mission execution controller: The mission execution controller is primarily responsible for differential control of the progress of the unmanned platforms with respect to the mission plans provided at the UI. The mission execution controller evaluates the robot's state against the mission plan and provides the command and control UI with appropriate feedback mechanisms. The mission execution controller is responsible for maintaining the current mission state and sequencing the subsequent, desired states based on the mission plans provided by the MPCS. Excessive deviations from the mission plan or state requires replanning, and this results in a new mission plan request to the MPCS.

Command and control UI: This UI provides the primary front end for user which includes all the tools necessary to monitor and control the robots [28]. Several information-rich sensors mounted on the robots such as ToF, RGB, IR and stereo cameras will be used to improve the performance in search and rescue tasks. The command and control UI provides the main map/crisis data viewing capabilities to enhance the robot operator's situational awareness of the SAR mission including progress of robots and first responders in the field. The UI presents the data generated by the mission execution controller to determine the mission-level progress of the robotic platforms. The UI will provide commanding capabilities for the UAVs, UGVs and USVs (abstracted by the level of autonomy). The commanding capabilities provided by the UI will include joystick inputs, spatial waypoints and mission-level commands (if supported by the platform). The UI interfaces with the platform command manager to deliver the commands to the robotic platforms. The command and control UI will rely primarily on touchscreen, keyboard and joystick inputs. An additional input device in the form of the exoskeleton will also provide a subset of command generation capabilities for the robotic arms mounted on the UGVs. The mission plans are GIS layers describing the sequence of tasks that must be performed for a given mission scenario. These plans are accessible by the mission execution controller. The mission plans are outputs of the MCPS system and are when available pushed to the RC2 mission plan database through the MPCS synchronizer.

Sensor fusion algorithms: This component will provide a set of algorithms for multi-robot multi-sensor data fusion. The command and control module can receive raw and on-board preprocessed data from the different robots. Under certain conditions and when the command and control module requests so, the sensor fusion algorithms are responsible to post-process this data provided by the data manager and translate it into a consistent representation usable by the rest of the components. The sensor fusion algorithms can act at different abstraction levels: robot states (i.e. health, navigation state), imagery, maps, features and landmarks.

GIS server and synchronizer: This component is the repository where the system will store all geospatial data gathered for the different components of the system. This component allows transforming the geospatial information storage in the system to the appropriate format allowing map viewers to compose this information in a final map. This component uses different OGC services [web map service (WMS), web feature service (WFS), web feature service—transactional (WFS-T)] for synchronization (upload and update) between the

information storage in the system and information gathered from the mobile devices at the RC2s and the between the MPCS and RC2s.

Mobile device server: The field device manager handles the data flow from the various mobile devices in the field. Its purpose is to handle and route text message flows and map updates and latest crisis data between the RC2 and mobile devices in the field. It will remain the central system to pull location data from the mobile devices, i.e. device GPS position. The component will use XMPP/Jabber standards for instant messaging support. In summary, the field device manager will ensure connectivity between the field devices and the GIS on the MPCS and RC2.

Communication interface: The communication interface manager is the middleware responsible for managing all data communications between the various actors in the crisis area (R2C, MPCS, Robots, etc.). The communication manager will implement data streams that provides access to the different data uplink and downlink to robots, ensuring that link quality and loss handling are adequately covered according to the requirements necessary for the application (sensors, video, etc.). The application programming interface (API) offers interfaces to encapsulate the traffic requested by applications within ICARUS communications framework.

3.2.3. Data fusion module

This module, in combination with the C2I user interface, has been designed to help the operator to get a clear overview of the emergency situation [29]. The following list shows a simplified concept of operations workflow from the initial reconnaissance flight to the development of the mission (also depicted in the figure below). In **Figure 13**, we can see the different functionalities describing the data fusion module as follows:
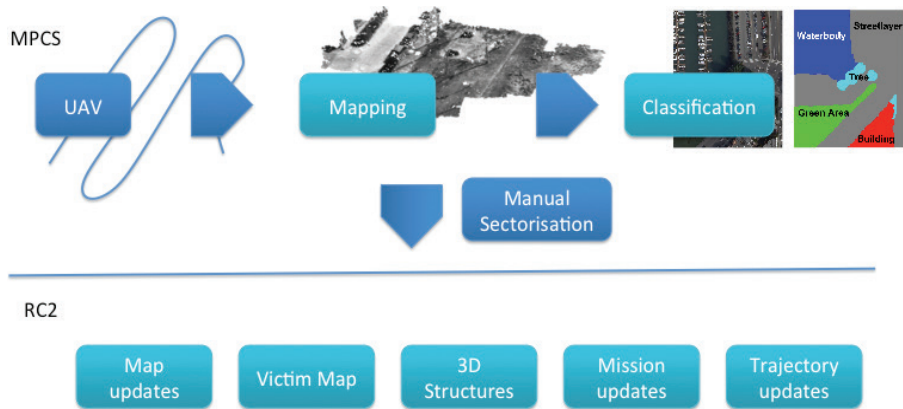

Figure 13. Concept of operations with data fusion functionalities (source: ICARUS).

1. From the MPCS, the initial high-altitude flight with the long-endurance UAV is launched.

2. This gathers an initial set of high-altitude (and presumably low accuracy) images that are used in data fusion to create the initial map of the area.

3. This map is used to show the operator the current state of the area of interest.

4. In parallel, this map image is parsed through a surface contextualization (characterization) that proposes sections between concepts such as forest, water, buildings, roads, etc.

5. The operator, with the help of points (1)–(4), has a general overview of the situation and can manually create sectors that will be distributed through the different RC2.

6. Each RC2 will be given a sector to start the operations, with the initial map done in (2).

7. The operator in RC2 will then ask for higher-accuracy and lower-altitude images on specific areas to update the map with visual images, possible location of victims, 3D structures, GIS updates, etc.

The specific architecture of this module and its interaction with other modules (namely command and control UI and geospatial database) is illustrated in **Figure 14**. As general comments, the module will be implemented in C++ with the possibility of integrating ROS in order to ease testing and scenario replay during implementation. In the final version, direct read and write to the database might be the chosen approach to gather the required information to build up the results and storage of the resulting images and GIS updates. The big picture of the data fusion architecture is summarized in the following picture.

A state of the art description along with the proposed approach to develop each functionality (each box in the previous picture) is described in the following subsections.
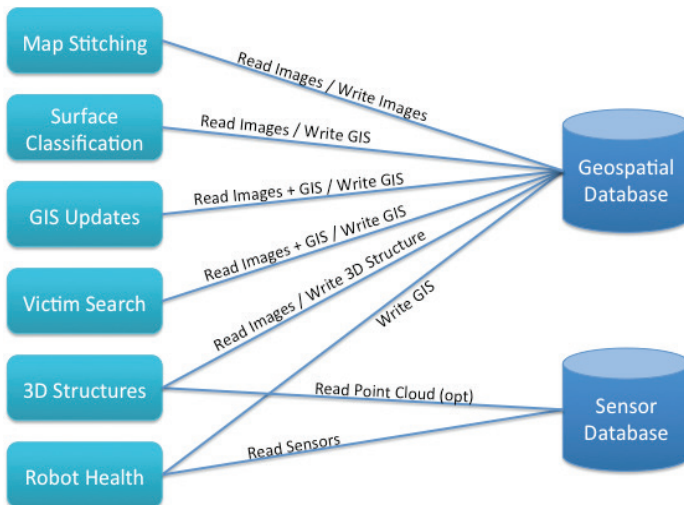

Figure 14. High-level data fusion architecture (source: ICARUS).

3.2.3.1. Map stitching

For this approach, the main key points of the object (image to stitch) will be detected and extracted along with the ones of the map. The surf feature detector and surf descriptor extractor will be used for that step. Other descriptors are being considered depending on the time and quality demands of the end-user. The descriptors will be computed and then matched using the Flann based matcher. Notice that other matchers like the brute force can be used too. Once the matches are computed, they will be used to get the homography function letting us to wrap the object on the same plane as the map and attach them in the same image. After an evaluation of the approach, OpenCV seems to be a good choice for the image computing library.

3.2.3.2. Surface classification, GIS updates and victim search

In this step, the main objective is to extract as much information as possible from the UAV's images. The type of terrain is going to be computed using a grid of surf descriptors applying a threshold. This segmentation will suffer two steps of optimization: first of all, small segments will be connected or erased; secondly, the regions will try to grow and see if colliding terrain can be added. If so, a texture and colour classification process will decide which type of terrain the conflict region is most probably in.

3.2.3.3. Map segmentation

The classifier proposed is support vector machine (SVM), which uses learning algorithms that analyse data and recognize patterns. During the training algorithm, SVM builds a model that assigns new samples into one region or other, making it a non-probabilistic binary linear classifier. An SVM model is a representation of the samples as points in space, mapped so that the samples of the separate regions are divided by a clear gap that is as wide as possible. New samples are mapped into the same space and predicted to belong to a region based on which side of the gap they fall on. The classification is based on colour image, where each pixel of the map (samples) is classified by its value of hue, saturation and value (HSV). Based on that premise, the red, green and blue (RGB) colour of the original map is converted to HSV. Hue defines the shade, which means the location in the colour spectrum (the neutral colour) that is determined by the reflective property of the object surfaces and it is relatively stable. Saturation describes how pure the hue is with respect to a white reference. Value defines the brightness, amount of light that is coming from the colour. These two depend on occlusion variation and the shape of the object.

The RGB colour of the map not only depends on the camera configuration (focus, exposure, lens, etc.) but also on the weather conditions (i.e. Sun elevation and clouds that may vary the brightness). Based on these premises, the classifier needs to be trained with the desired regions (vegetation, land, water, etc.) what is known as ground truths; the user must determine a small but representative set of pixels for each region. At this point, the classifier builds a model that may be used to classify the maps.

The SVM prediction is implemented in a ROS service; when the service is called, the original map is taken from a specified path of the hard disk. The map is divided in several areas; the

total amount of areas is the same as cores have the computer where the service is called. A multithread is launch to classify (predict) the entire map minimizing the computational time. The prediction process takes normally around 2 minutes. Finally, the segmented map is saved in another specific path of the hard disk. The entire procedure is summarized in the following flow chart (**Figure 15**):

3.2.3.4. Map generation

The objective of this module is the creation of a 2D aerial map in near real time. This map is produced from the images provided by the different aerial robots, and its main purpose is to furnish the operator with a quick update on the conditions of a particular patch of terrain. Additional maps can also be produced in the post-processing step such as a digital elevation model (DEM) and a 3D structure (in form of point cloud or mesh).

First of all, the key points are detected and extracted for every image and stored in their respective keyfile. As soon as an image keyfile is ready, its key points are matched with the ones of the previous images. During this stage an optimization using the GPS coordinates allows us to reduce the number of image comparisons by more than a 90%. This fact also allows us, most of the times, to run the matching process in near real time. At the end of the matching, we use the matching table to perform a bundle adjustment and retrieve a 3D sparse
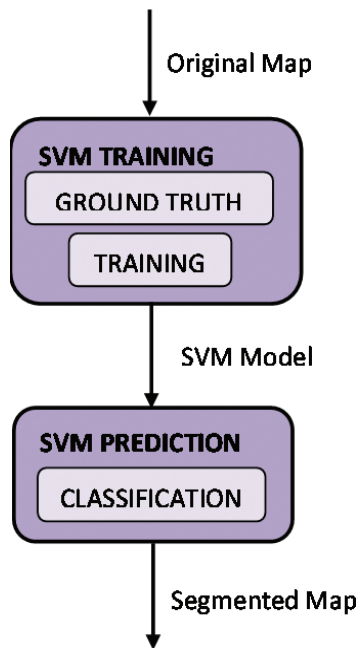

Figure 15. Map segmentation box diagram (source: ICARUS).

point cloud. Once the 3D point cloud is ready, we use it to create a 2D projection or a 3D render depending on the user demand. This pipeline is depicted in **Figure 16**.

3.2.4. Automated mission planner

Mission planner is a stand-alone module of the C2I designed to be a support tool during the action-planning phase [30]. The planner facilitates the preparation of a mission plan for each team and sector. Data form the MPCS database is used for this purpose. Mission planner has two main elements: symbolic planner and specialized planners.

3.2.4.1. Symbolic planner

The symbolic planner (or 'task planner'), **Figure 17**, is the core component of the toolset supporting the ICARUS mission planning. It is part of the MPCS and is therefore running in the OSOCC. The purpose of the symbolic planner is to generate detailed action plans for the ICARUS' robots, accounting for the mission context and available information on mission progress. The symbolic planner, as its name means, takes as input (1) a symbolic representation of the knowledge about the mission (environment, mission context, available resources, various constraints including temporal ones, etc.) and (2) high-level mission objectives (goals) expression. The planner generates one (or several) task plan(s) that can be handled at the RC2 level for a coordinated execution by the different robots (relying on the RC2's mission execution and control manager). The symbolic planner relies on a LISP implementation of the Shop2 HTN planning engine, exploiting a hierarchical definition of the planning domain. As per this paradigm, high-level methods are decomposed into lower-level tasks (either methods or operators—in blue, in the pictures below) when method's preconditions are satisfied, until the planner reaches primitive tasks. We moreover introduce in the planning scheme time considerations thanks to an encoding of the domain exploiting the so-called multi-timeline processing (MTL). This scheme allows expressing durative and concurrent actions and allows effectively accounting for time constraints.

As part of the planning scheme, we introduce specific operators that allow performing on-the-fly (i.e. during the planning process) requests to the specialized planners—this deals, e.g. with estimation of time or energy consumption for navigation between two points in the environment or for the identification of best suited location to perform perception. Results from queries to
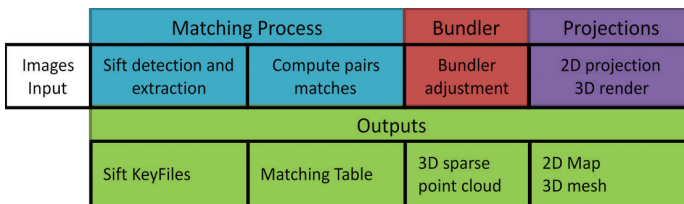

Matching Process		Bundler	Projections
Images Input	Sift detection and extraction	Compute pairs matches	Bundler adjustment	2D projection 3D render
Outputs			
Sift KeyFiles	Matching Table	3D sparse point cloud	2D Map 3D mesh

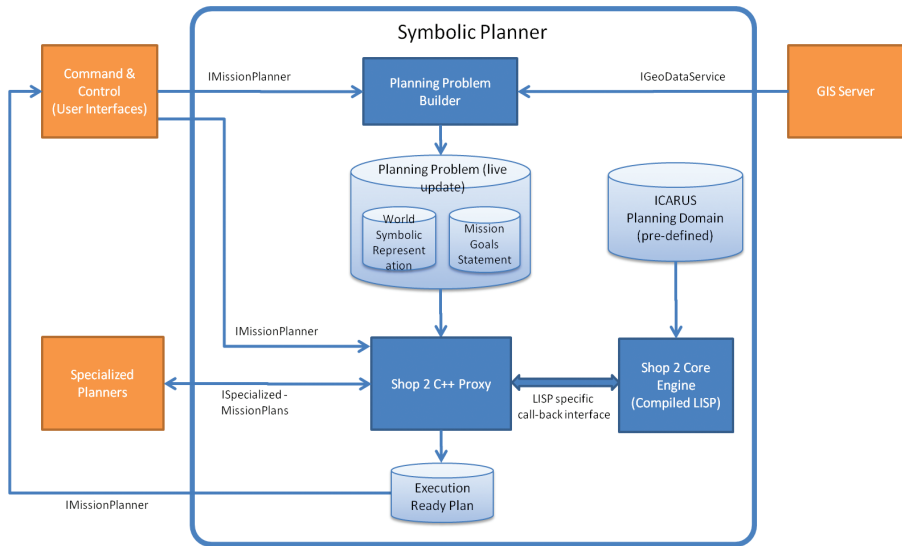**Figure 16.** Map generation box diagram (source: ICARUS).

Figure 17. Symbolic planner architecture (source: ICARUS).

the specialized planners are considered in the generated task plan, accordingly. We summarize in this section the components and their connections as part of the symbolic planner, as it is implemented for the MPCS. The symbolic planner basically consists of the three following components:

1. The Shop 2 Core Engine is the planning engine, which is based on the Open Source Shop 2 planner (LISP implementation). It takes the ICARUS planning domain and the live update of the planning problem as inputs that consist of (i) the symbolic representation of the world and (ii) the mission goals statement.

2. The world symbolic representation and the mission goals statement are formatted in the proper planning formalism through the planning problem builder (C++ implementation). This component requests information about the actors and ongoing mission situation and maps data that are relevant for the planning process. This includes models of the available resources (robot, personnel, etc.) and status of these resources (power left, availability, etc.) All this information is obtained from the GIS Server. The mission goals statements are obtained from the command and control system, with a dedicated user interface for mission definition.

3. As a mean to interface conveniently with the Shop 2 Core Engine (which, as mentioned before, is LISP based), a Shop 2 C++ proxy allows interfacing in a conventional manner with components that interact or may have to interact with the planning process—mainly (i) the specialized planners that supports the symbolic planner during the planning process with

specific planning capabilities requiring, e.g. semantic or motion/path planning-related evaluation, and (ii) the command and control interface, from where the planning process is handled (e.g. starting new planning cycle, modifying planning policy or parameters, etc.). This proxy should also turn rough task plans, as generated in the Shop 2 planner formalism, into an execution-ready plan that complies with RC2 formalism expectations (through the command and control interfaces) and that the RC2 can therefore directly exploit.

3.2.4.2. Specialized planner

Specialized planners form a module that responds to requests from the symbolic planner. The requests concern detail, computation heavy problems such as path planning, proper positioning, etc. Specialized planners use a semantic model of the environment (SME) constructed by a subsystem of the planners based on the GIS and data gathered by the unmanned platforms.

The specialized planner module consists of two main parts: semantic environment constructor and query processor. The semantic creator gathers data from GIS server and sensor fusion feed and analyses them to create the SME representation of a given area. The creator performs basic concept recognition according to a defined ontology. Query processor works as a server. The client sends a query, which defines the task and provides needed parameters. The processor then tries to formulate a response based on the SME model and given parameters. The query response is then sent to the client. The planners use specialized technologies to improve computation time and SME creation:

- NVidia PhysX: This popular physics engine is used to simulate the SME. It allows for simulating concepts in form of static and dynamic entities and provides tools for automatic event catching and handling. The events are used to follow the relations between concepts.

- NVidia CUDA: This SDK allows to perform parallel computation on graphical cards. This allows a decrease in computation times for many parallelizable algorithms.

The planners are being designed to work with a set of standards to provide consistency and compatibility with other C2I components:

- Qualitative spatio-temporal representation and reasoning (QSTRR) framework. It provides the base for the SME creation defining basic ontology.

- ROS: The module of the mission planner will be prepared as nodes of the ROS framework. This will provide means for easy communication with the rest of the C2I.

- QT: A popular set of libraries for creating GUI and application backend logic. The program will use QT classes for internal communication.

- OpenCV: Libraries for machine vision.

An important standardization element of the planners is ontology. It defines the concepts of the semantic model, relations between them and rules for maintaining integrity of the model. The next paragraph will show a short overview of the ontology.

Specialized planners consist of modules shown in **Figure 18**:

- Data reception and preparation module: This module is responsible for receiving the input data and preparing it to be used for SME creation. In the process, the data is grouped into packages. Each package contains information about single sector. Additionally, the data is being preprocessed, for example, 3D point clouds are filtered and normal vectors are computed for each point.

- Semantic model creation and upgrade module: This module is responsible for creating the semantic model of environment and distributing it to other modules. Input data is processed to extract semantic information and transformed into the ontology-compatible format.

- Semantic model modification module: This module receives the queries from the symbolic mission planner and creates instances of the semantic model based on the received parameters. This process includes changing practicability of area considering robot type, including the sensor model.

- Main reasoner: This is the main reasoning engine for the specialized mission planner. The base for the module is PhysX-based simulation environment. The module creates a hypothesis space and then tests the hypothesis by a set of conditions. The hypotheses that are considered best are sent as an output.

- Secondary reasoner: Secondary mission planner reasoner is a module that answers special inner queries asked by the main reasoner. The advantage of this module is that it uses CUDA-based algorithms which allow for reducing the computation times.
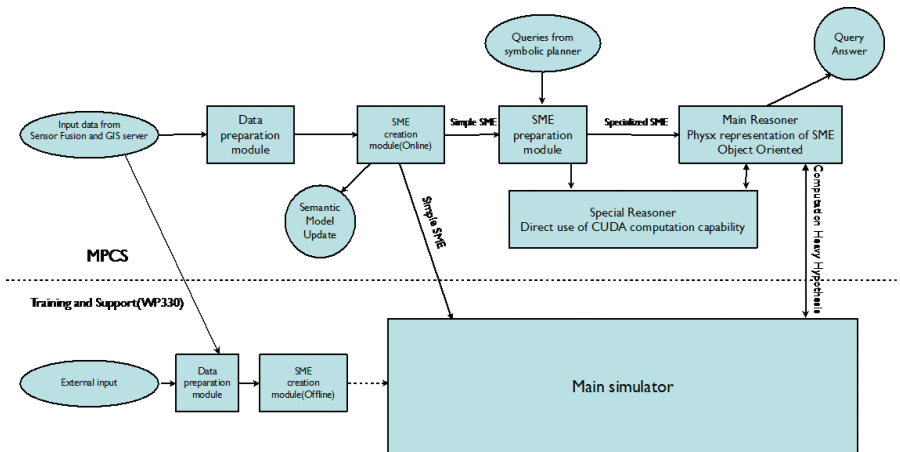

Figure 18. Architecture of the specialized planners (source: ICARUS).

3.2.5. GIS repository

3.2.5.1. Overview

The MPCS GIS repository is the main repository within ICARUS system, and it is typically located within the OSOCC infrastructure. Before the deployment of ICARUS system in the catastrophe area, the MPCS GIS repository is loaded with all cartography, imagery and thematic datasets related to that area, which will be used as input by the users (e.g. visualization of maps in the main workstation operated by the operator on duty) and subsystems connected to it (e.g. mission planner) to carry out their assigned tasks (e.g. locate with the support of robots, victims nearby crumbled buildings). The access and management of the information in the GIS repository are done through OGC standards and compliant http services by using POST and GET requests.

Apart from the local datasets stored within it once the system has been deployed, additional sources of information that might be of interest/support for the SAR operations through the access to external mapping services and information repositories (e.g. GDACS), providing thus complementary and useful information that can be used to improve ICARUS operations on the field. To that end, the MPCS provides a component in charge of dynamically accessing to these external sources of information and adapting it to ICARUS GIS repository internal data model based on humanitarian data model (HDM). In order to accomplish this, the component defines for each external service or repository a data model mapping, which describes how to transform the original data source into ICARUS internal data model.

In turn, at the beginning of each SAR mission, different geographical subsets of the MPCS GIS repository are copied locally to the GIS repositories within the different RC2 systems operated by the SAR teams in different areas. At the end of the day, the updated/modified information within the RC2 GIS repositories is synchronized and merged with the main GIS repository in the MPCS.

The aim of the RC2 GIS component is to store all the necessary information that the SAR personnel operating the RC2 component might need in order to accomplish their assigned tasks. In this regard, the RC2 GIS can be seen as a reduced version of the MPCS GIS, hosting a subset of the geographical layers and information contained in the MPCS GIS repository. During a mission, the RC2 GIS will update locally the original information by modifying its contents (e.g. the location of a victim) or adding additional resources (e.g. sensor information retrieved from the robots and stored in the RC2 repository, mobile phone images, etc.). At the end of the day, the local RC2 GIS repositories will be merged and synchronized with the MPCS GIS repository to update the central repository and have a homogeneous and coherent situation status for planning future missions. RC2 GIS repository will also store the mission plans sent by the MPCS, as well as any modifications that can be made locally if necessary.

Other important differences with the MPCS GIS are:

• RC2 GIS has no direct access to the external repositories, but if necessary it could access the retrieved data through the HTTP interfaces available in the MPCS GIS.

- Sensor data from robots (except for the case of the UAVS) are stored in the RC2 GIS reposi-tories and synchronized to the MPCS GIS (due to the bandwidth constrains for transferring large amounts of data).

The aim of the mobile device directly connects to the GIS server hosted on the RC2 via Wi-Fi and cache important WMS and WFS layers for offline operations, thus supporting the per-sonnel working on the field over the course of the mission execution. Due to the inherent limitations in the storage and computational capacity of this type of devices as well as with the related network bandwidth limitations which prevent from transferring large amounts of information between the RC2 or MPCS and the mobile devices, the approach followed by it differs slightly. The mobile device will store a basic set of layers, allowing the user to work offline and carry out typical operations such as updating information (e.g. set a building as visited, changing the location of victim to a new GPS coordinate, etc.) or creat-ing new resources by taking geo-tagged pictures with the mobile device camera. Once the user enters an area with network coverage (e.g. 3G or Wi-Fi), the mobile device GIS auto-matically will try to contact the RC2 GIS services to retrieve possible updated layers (e.g. using the WMS or WFS) and then update accordingly its local cache. In addition to the GIS repository, the mobile device GIS will also provide a user interface—based on HTML5 and JS technologies—that supports the user with the necessary functionality to manage and interact with the locally stored information. Typical operations available are (i) zoom in and out; (ii) pan; (iii) draw polygons and associated information to it; (iv) take geo-tagged images with the camera, notes, points of interest, etc.; (v) send and receive text messages; and (vi) connect and retrieve/provide information from RC2 and MPCS services (i.e. OGC and ICARUS legacy RESTful services).

3.2.5.2. Technologies and Standards

Table 1 presents the selected open source implementations for each of the databases and services mentioned above.

3.2.5.3. GIS architectures for MPCS and RC2

The aim of the GIS database component is to serve as a repository for storing, accessing and manipulating all the required geographical information used or generated in the context of ICARUS operations, thus a central part of ICARUS architecture. In this sense, several components and subsystems rely on the information it contains, such as the mission planner, the data fusion algorithms or the teams deployed on the field, which might require cartographic and aerial layers of the area where they are working in terms of maps or alphanumeric information. The GIS database is an integral part of the MPCS and RC2 subsystems.

It provides the same core functionalities for both with some specific differences regarding the requirements of those two subsystems. As mentioned before, the GIS repository will store dif-ferent geospatial layers, maps and any other information geospatially tagged piece of infor-mation by means of:

Component/service	Open source implementation	Description
Spatial database	PostgreSQL + PostGIS	PostgreSQL is an open-source object-relational database management system (ORDBMS). It supports a large part of the SQL standard and offers many modern features such as complex queries, foreign keys, triggers, updatable views, transactional integrity and multi-version concurrency control
		PostGIS is a spatial database extender for PostgreSQL object-relational database. It adds support for geographic objects allowing location queries to be run in SQL. In addition to basic location awareness, PostGIS offers many features rarely found in other competing spatial databases such as Oracle Locator/Spatial and SQL server
OGC WMS	GeoServer/MapServer	GeoServer is an open-source software server written in Java that allows users to share and edit geospatial data. Designed for interoperability, it publishes data from any major spatial data source using open standards
		MapServer is an open-source geographic data rendering engine written in C. Beyond browsing GIS data, MapServer allows you to create 'geographic image maps', that is, maps that can direct users to content
OGC WFS	GeoServer/MapServer	MapServer only supports read-only operations in the WFS interface. For update operations, we will use GeoServer
RESTful interfaces	Apache CXF	Apache CXF is an open-source service framework. CXF helps building and developing services using front-end programming APIs, like JAX-WS and JAX-RS. These services can speak a variety of protocols such as SOAP, XML/HTTP, RESTful HTTP or CORBA and work over a variety of transports such as HTTP, JMS or JBI. Within the context of ICARUS, it will be used to implement the ICARUS legacy RESTful interfaces to manage and access the geo-resources
Web application server	Apache and Apache Tomcat	The services mentioned above will be run in Apache web server and Apache Tomcat (web application server)
Spatial database	SQLite	SQLite is an in-process library that implements a self-contained, server-less, zero-configuration, transactional SQL database engine. The code for SQLite is in the public domain and is thus free for use for any purpose, commercial or private

Component/service	Open source implementation	Description
User interface and map client	HTML5 + OpenLayers 2.0 + GeoExt + ExtJS	In order to make the mobile device deployable in a wide range of device platforms (i.e. Android, iPhone, etc.), it will be based on a set of standard and open-source-based components
		HTML5: It includes detailed processing models to encourage more interoperable implementations; it extends, improves and rationalizes the mark-up available for documents and introduces mark-up and application programming interfaces (APIs) for complex web applications. For the same reasons, HTML5 is also a potential candidate for cross-platform mobile applications. Many features of HTML5 have been built with the consideration of being able to run on low-powered devices such as smartphones and tablets
		OpenLayers: It is a pure JavaScript library for displaying map data in most modern web browsers, with no server-side dependencies. OpenLayers implements a JavaScript API for building-rich web-based geographic applications, similar to the Google Maps and MSN virtual Earth APIs. Furthermore, OpenLayers implements industry-standard methods for geographic data access, such as the OpenGIS Consortium's web mapping service (WMS) and web feature service (WFS) protocols. As a framework, OpenLayers is intended to separate map tools from map data so that all the tools can operate on all the data sources
		GeoExt: GeoExt brings together the geospatial know how of OpenLayers with the user interface savvy of Ext JS to help building powerful desktop style GIS apps on the web with JavaScript
		ExtJS: Ext JS brings a rich data package that allows developers to use a model-view-controller (MVC) architecture when building their app. The MVC leverages features like Big Data Grids enabling an entirely new level of interactivity in web apps

Table 1. Overview of GIS services and standards used.

- Files (typically for raster images such as GeoTIFF, JPEG, point clouds, ESRI shapefiles, etc.).

- Relational spatial database (typically for vectorial and alphanumeric data).

- HTTP RESTful services compliant (in most cases) with OGC standard interfaces and operations in order to make it interoperable with other external services and subsystems (e.g. mobile device used by field teams accessing to the latest aerial images located in the RC2 GIS repository through the OGC WMS service). Using the OGC standard interfaces, a set of supplementary operations provide additional functionalities not covered directly by these standards, such as the upload and management of dynamically generated geo-resources to the ICARUS GIS repository (e.g. sensor data, mobile device images, geo-referenced text messages, etc.).

Currently the architecture in **Figure 19** includes some geospatial information systems (GIS) standard services based on open geospatial consortium (OGC):

- Web map service (WMS): It serves geo-referenced map images, and it supports pyramidal raster; an image pyramid is several layers of an image rendered at various image sizes, to be shown at different zoom levels. Main operations performed by the service are:

  ○ GetCapabilities

  ○ GetMap

  ○ GetFeatureInfo

- Web feature service-transactional (WFS-T): It is capable of serving features, and it allows creation, deletion and update of features. Main operations performed by the service are:

  ○ GetCapabilities

  ○ DescribeFeatureType

  ○ GetFeature

  ○ Transaction (update, insert, delete, edit)

- Styling: The maps from the WFS service have customized styling; this is done with styled layer descriptor (SLD) technology for all open street map data. The rest of WFS data depends on the client side.

The software components in **Figure 20** include the deployment and configuration of two main components in addition to PostgreSQL database:

- Tomcat 7 is a servlet container supporting 52 North SOS and GeoServer as well as GDACS services. The main components deployed on it are:

  ○ GeoServer: This is a java-based service deployed under Tomcat 7. Its purpose is to act as WFS-T and WMS. Its main advantage is that it provides transactional operations over the vectorial data within the database.

  ○ MapServer: This is a C-based service deployed under Apache 2 as a CGI, and its capabilities are to work as WFS to provide different output format responses apart from Geographic Markup Language (GML); indeed this service response could be a CSV or a JSON file. As WMS, it supports Enhanced Compression Wavelet (ECW) raster format.
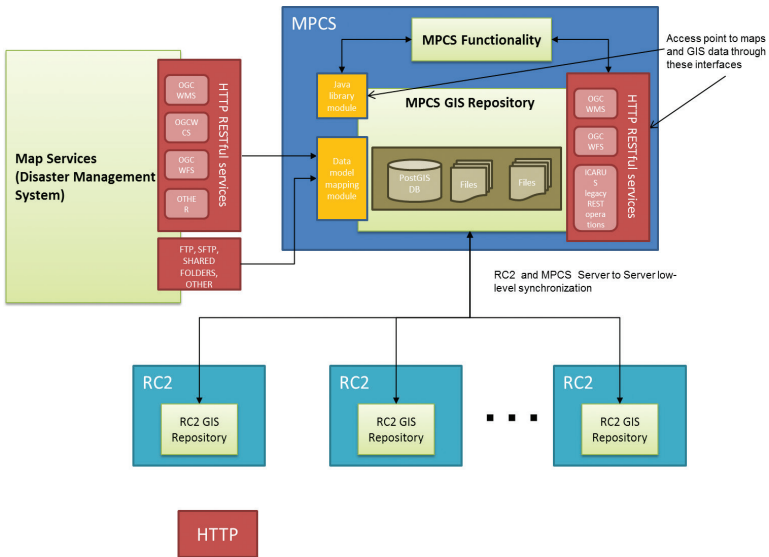
Figure 19. MPCS GIS high-level architecture (source: ICARUS).
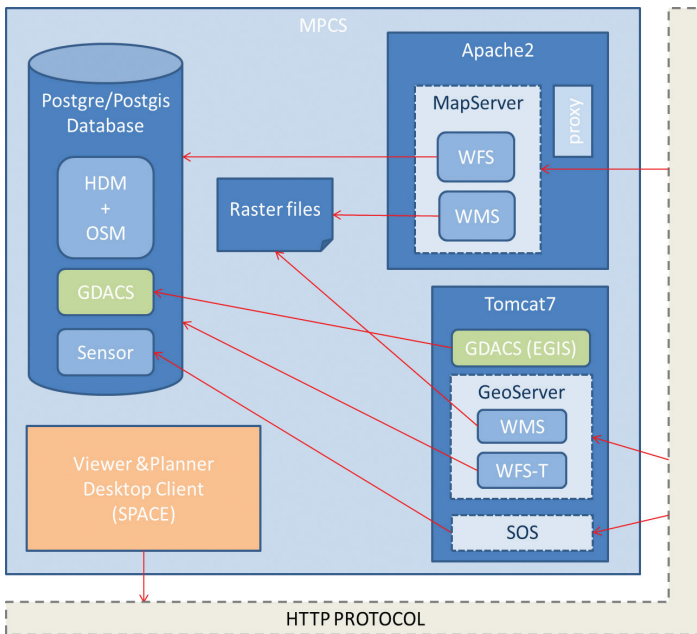

Figure 20. GIS software components (source: ICARUS).

- Apache 2 web server has been configured to provide Common Gateway Interface (CGI) support to make MapServer working, and it is also the main entrance to the server through port 80 and redirects all traffic to Tomcat 7.

  ○ The Apache 2 web server oversees publishing sensor images stored in the system. This server has installed Python library, and it is configured to support a Python-based proxy to allow usual third-party javascript requests.

There is a Postgre databases already installed and extended with PostGIS to support all the geospatial functionality. The ICARUS schema is composed of:

- Open street map (OSM) tables, storing vectorial data for Lisbon, Moia and Marche-en-Famenne. For each scenario, there are three tables (polygons, points and lines). Those tables have been expanded with several columns to match humanitarian data model (HDM) schema.

- Internal ICARUS tables to keep track of mission, its zones and sectors, as well as teams and its members (humans or robots) as well as their positions through the waypoints table. There are structures and victims that could be located, since, apart from specific data, all these tables have a geometry field to be able to geospatially locate each occurrence.

3.2.5.4. External crisis data

The purpose of integrating map layers from external suppliers is to have a greater amount of information, that is accurate and up to date. The integration of information from other crisis management systems will permit to release systems and other resources partially of workload, without losing functionality. In certain cases, external data sources will allow comparing external information with GIS internal information, obtaining more detailed information. Comparing internal information makes it possible to obtain a more complete picture of the situation.

3.2.5.4.1. Global disaster alert and coordination system (GDACS)

The global disaster alert and coordination system (GDACS) provides near-real-time alerts about natural disasters around the world and tools to facilitate response coordination, including media monitoring, map catalogues and virtual on-site operations coordination centre. GDACS (**Figure 21**) is a web-based platform that combines existing web-based disaster information management systems with the aim to alert the international community in case of major sudden-onset disasters and to facilitate the coordination of international response during the relief phase of the disaster.

GDACS provides the 'virtual OSOCC' (www.gdacs.org/virtualOSOCC) to coordinate international response. The virtual OSOCC is restricted (password protected) to disaster managers worldwide.

- GDACS information service providers are organizations or services that provide or manage disaster information. These include:

- European Commission Joint Research Centre: Automatic alerts and impact estimations

- OCHA/virtual OSOCC: Web-based platform for real-time information exchange among disaster managers

- UNOSAT: Provision and coordination of map and satellite image products

- OCHA/ReliefWeb: Repositories of damage maps and impact analyses, which in the aftermath of a disaster are made available through an RSS-based catalogue, which is available in GDACS

GDACS information service providers share information and synchronize their systems according to GDACS data coordination standards. These are:

- Extended really simple syndication (RSS) feeds to transfer and integrate information between databases and websites of its users.

- The GLIDE number (www.glidenumber.net) as unique identifier for disasters to link information related to a given disaster.

- Common Alerting Protocol (CAP).

3.2.5.4.2. MapAction

MapAction is an international NGO that provides maps and other information services to help humanitarian relief organization in field. They are responsible for the data collection and information management and also offer access to mapping information (in paper and digital format).

3.2.5.4.3. Software architecture

The most important thing is to perform an initial analysis of the generic structure of the GeoRSS that is going to be integrated. It is essential to know the refresh rate of the selected external provider data sources. If the refresh rate is variable, it is needed to define a parameter that sets the time interval in which to check for updates have occurred in the source. GDACS implements a system of email alerts; it might be possible to detect these warnings and proceed to check if there is an update in the data. Subsequently it is necessary to
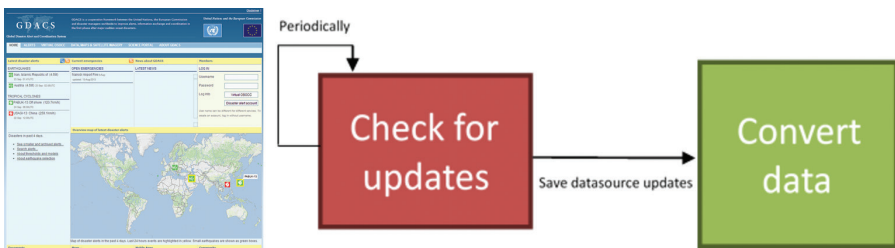

Figure 21. (i) Homepage of GDACS, http://www.gdacs.org/. (ii) Periodical update of GIS with data (source: ICARUS).

compare the data structure of the original source and see how the information can fit in the data model of the developed system. Consequently, a process responsible for periodically checking for updates in the data sources will be created. If an update has occurred, data will be retrieved. A system based on predefined rules from the previous studies will be developed; retrieved data that has been collected will be converted to the data structure defined in the application.

Stored data are in the following tables in PostgreSQL:

- Gdacsitem: current disaster items (RSS last reading data)

- Gdacsitemhist*: all historical items

- Gdacsresource: resources associated with the item

The most relevant data are collected from the following RSS:

Disaster items:

- Identifiers: unique disaster identifier + episode identifier

- Registration data in our system and item data

- Title of disaster and description

- Alert level and description of the magnitude

- Event type

- Country

- Affected population and victims range

- Position (latitude and longitude): geometry

Item resources:

- Item identifiers and the episode with which it interacts

- Register data in our system

- Title and description

- Resource source

- Link

- File type (image/wms/xml/txt)

RSS reading: RSS reading is done in an ongoing basis. A thread has been built which reads the RSS, and it compares the changes with the last reading existing data. In this way, it only registers new items, and it withdraws those who are not active. The development consists on a JAVA web service that has the ability of configuring the most suitable reading interval.

Files: Associated to items, there are many resources such as documents, images etc. that can be accessed through URLs. The web application that reads the RSS, in addition to store data into the database, stores files (**Figure 22**) locally of those resources that we are interested in and that may be imported. For instance, a URL of a WMS does not help us and due to this: a configurable white list has been created with those resources extensions in which we are interested.

Layers and symbolization: Stored data in GIS database as seen in the GDACS-GIS architecture (**Figure 23**) have the geographic localization of the disaster (lat and long). Both items table as historical items are published through the GeoServer map server. The two published layers are symbolized in the same way as in GDACS website. So as to that, we have a Styled Layer Descriptor (SLD) and an array of icons to represent different states and disaster types. Disaster items are depicted by the value of the field 'subject'.

3.2.5.4.4. Merging into GIS

By comparing the GeoRSS catalogues of GDACS and MapAction, the latter has a smaller amount of information. Another reason to decide that GDACS is going to be the main external data provider is that it has a clearly predefined structure for the GeoRSS catalogue. This standardized structure will facilitate the automation of the integration of external data into ICARUS data model. GDACS has the following standards to publish information:
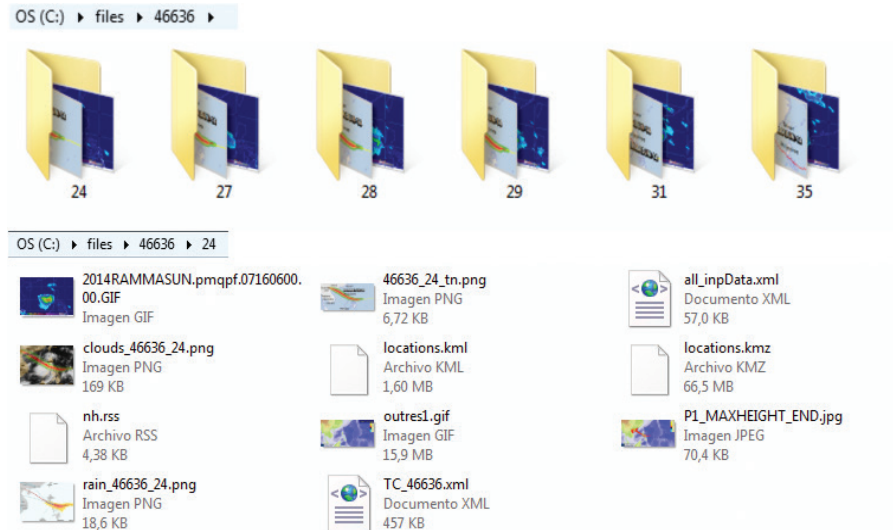

Figure 22. Disaster episode folders and files in each episode stored in file system (source: ICARUS).
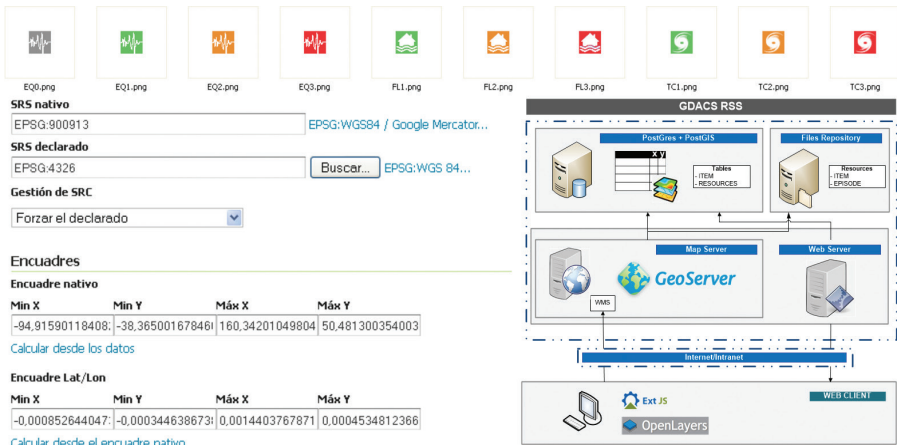
Figure 23. GDACS-GIS architecture (source: ICARUS).

- Feeds must be compatible with all RSS and GeoRSS viewers.

- The main GDACS feeds must contain links to all GDACS partner feeds, allowing applications to drill down to more information.

- Model results must be made available as a separate feed. However, key data can be exposed in the main GDACS feed.

- GDACS main feed must contain a minimal set of standard GDACS elements that are available for all disaster types. These must be compatible with CAP for easy transformation:

  - Time (period): from, to and status (forecast, ongoing, ended)

  - Information on whether event is 'active'

  - Event type

  - Severity: abstract independent of hazard but containing enough information for characterizing the full severity

  - Population in affected area

  - Vulnerability of affected country

  - Alert core/level

  - Severity (CAP)

  - Urgency (CAP)

  - Certainty (CAP)

- An identifier section disambiguates many identifiers.

- A resources section lists all GDACS partner information feeds.

Figure 24 shows the structure of an RSS file served by GDACS. As can be seen, there are a series of tags that define various attributes of the data source (title, description, access level) and finally the resource. In the example the data source is a type WMS.

3.2.5.5. HDM extensions for ICARUS

This section provides details on how to relate the HDM and the extensions provided above to the relational spatial database used to store and manage these layers. The GIS repository follows the humanitarian data model (HDM)—with additional extensions/adaptations necessary to fulfil ICARUS informational requirements—thus providing a common and interoperable data model shared among all applications and systems within ICARUS that requires geospatial information. This, in addition, has the advantage of allowing the integration of external data sources that comply with HDM as well as offering ICARUS information to external parties. Extensions of the HDM with layers which are of interest for ICARUS purposes are as follows:

Geographical sectorization: Subdividing a geographical area into several sectors is an important feature that the C2I system must have, to support asset organization, mission analysis, decision-making, etc.

```
      <gdacs:title>OCHA Situation Reports for Earthquakes</gdacs:title>
   ▼<gdacs:description>
       List of most recent OCHA situation reports published on ReliefWeb
     </gdacs:description>
     <gdacs:accesslevel>Public</gdacs:accesslevel>
   </gdacs:resource>
▼<gdacs:resource id="geooffice" version="0" source="INGV" url="http://cnt.rm.ingv.it/"
type="link">
     <gdacs:title>Istituto Nazionale di Geofisica e Vulcanologia</gdacs:title>
   ▼<gdacs:description>
       Latest earthquakes recorded by the Italian National Earthquake Centre
     </gdacs:description>
     <gdacs:accesslevel>Public</gdacs:accesslevel>
   </gdacs:resource>
▼<gdacs:resource id="gdacs_active_alerts" version="0" source="JRC"
url="http://dmarcgis.jrc.it/arcgis/services/GDACS/gdacsAlertsActiveBG/MapServer/WMSServer"
type="wms">
     <gdacs:title>WMS service of GDACS Active Alerts</gdacs:title>
   ▼<gdacs:description>
       This WMS service shows GDACS alerts of the past 7 days.
     </gdacs:description>
     <gdacs:accesslevel>Public</gdacs:accesslevel>
   </gdacs:resource>
▼<gdacs:resource id="gdacs_active_earthquakes" version="0" source="JRC"
url="http://dmarcgis.jrc.it/ArcGIS/services/GDACS/gdacsEQactiveBG/MapServer/WMSServer"
type="wms">
     <gdacs:title>WMS service of GDACS Earthquake Alerts</gdacs:title>
   ▼<gdacs:description>
       This WMS service shows the earthquakes in the past 7 days
     </gdacs:description>
     <gdacs:accesslevel>Public</gdacs:accesslevel>
   </gdacs:resource>
```

Figure 24. GDACS GeoRSS example, http://www.gdacs.org/XML/RSS.xml (source: ICARUS).

Strategic locations: This should be specified in the C2I filters.

Buildings: In catastrophes that happen in land, such as earthquakes, buildings can suffer different degrees of structural damage, from simple cracks in the walls to destruction. In such cases, often individuals become trapped inside buildings, and SAR operatives must enter these buildings in order to rescue the trapped victims. Important temporary sites

Victim recovery operation: Rescuing victims in any disaster scenario is one of the top priorities of any SAR operation and to maximize the efficiency of all the SAR teams on the field, the C2I must employ the necessary tools to ensure that all victims are tracked and assigned to a team.

Human and robot tracking: When SAR operatives are deployed on the field, each of them is assigned to a team. After teams have been formed, their members are then able to cooperate efficiently in rescue missions that are assigned to them.

Mission plans: When a location is identified as either having a possibility or certainty of having victims, a SAR mission is immediately created, associated with a search area and assigned to a SAR Team if one is available.

3.2.5.6. Low-level synchronization between MCPS and RC2

At the initial moment, both MPCS and RC2 GIS repositories contain the same version of the information. Over time the information in both components is modified locally (e.g. MPCS GIS receives new maps with additional features from external services, RC2 GIS repository is updated with new victim status or mobile photos are stored), and therefore they will be out of synchronization as it is difficult to make frequent online synchronization among them due to the network bandwidth constrains. Within ICARUS GIS repository, the relational database is used to store all the vectorial layers but also to link those geo-resources that are stored in the system (e.g. images uploaded from the mobile device, sensor data from the robots, etc.). In order to keep track of the changes in the different GIS repositories (both in the MPCS and the different RC2 available), the following approach has been taken.

Bucardo is an asynchronous PostgreSQL replication system, allowing for both multi-master and multi-slave operations. Bucardo is required only to run in one server and as such the MPCS was selected has host for the synchronization process due to its hierarchical relation to the other systems. After installation and configuration, Bucardo instals an extra layer on each synchronized database. This layer ensures that all data, even if there are connectivity problems, gets synchronized once all databases regain connection to the central synchronization service, in this case, hosted by the MPCS. Because all nodes in the synchronization service have permissions to write in the database, a multi-master relationship was used. When there is connectivity between all nodes and transferred amount of data is small, data replication across all nodes is almost real time.

Considering that Bucardo system will synchronize database tables of the different C2I's, a series of triggers have been set in database to ensure providing unique IDs to every database table. This was needed because usually GeoServer manages the feature ID generation of any new geometry added to the system and does not take this conflict into consideration.

3.2.5.7. Other support layers

Apart from HDM and the extensions provided for ICARUS, there exists a set of useful datasets (e.g. OSM, land, air and sea maps provided by RMA and other external data sources) that although not directly used as input for processing, they can provide further support to the different users for having an improved situation picture:

- Open street maps

- Land, air and sea maps

- Maps and layers from other crisis management systems

- MapAction and GDACS

3.2.6. Mobile interface

Figure 25 depicts the component architecture for the mobile interface. The different components are described below.
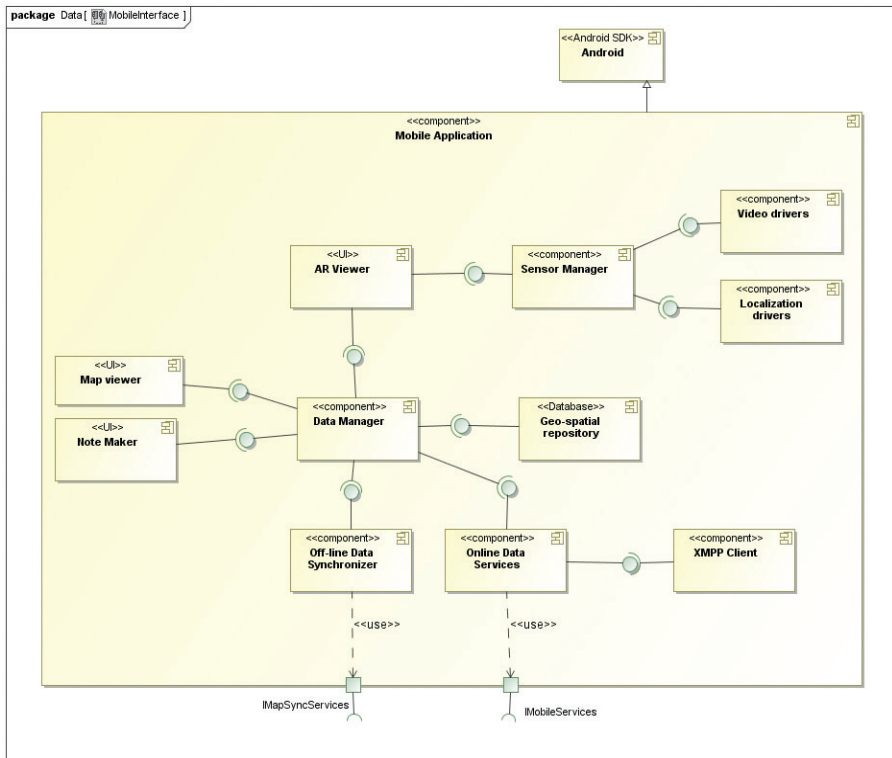
Figure 25. Component architecture for the mobile application (source: ICARUS).

Offline data synchronizer: This component allows the mobile devices to upload the system with the data gathered from the mobile devices on the field and vice versa; it allows updating the mobile devices with the information storage in the main system. The synchronization of the data has to be guaranteed without any type of net communication.

Online data services: This component is responsible for data sharing between the mobile application and the RC2. Two separate implementations are foreseen within this component, one focusing on text/voice message exchange and the other focusing on location data exchange. The component is primarily responsible for handling connections and data flow using the native android socket API. For the location data exchange, it will expose a socket for the data manager to share device location data and receive location data from other devices (mobile devices and the RC2). For text messaging, it will expose a socket for the XMPP client to send and receive text messages.

Data manager: The data manager is responsible for handling and distributing geospatial information. It services requests for geospatial data primarily from the map viewer and note components. As all data within the mobile application can be considered geospatial (including notes taken at a particular location), the data manager provides get/set methods for each of these UI components. It handles database read and write functionality and ensures that all geo-data is maintained in a consistent manner. In addition, the data manager maintains all communications with external data services.

Geospatial repository: This component allows to store geospatial information in the mobile device, allowing it to work either offline and online.

Map viewer: This component allows the end-user to see the geospatial information available for the system in a map viewer. In addition, this component provides the basic functionality (zoom in, zoom put, pan) for the navigation through the map.

Note maker: This component allows the end-user to introduce a note marker within the map. The end user can tap/click over the map at any location and this component provides a menu to setup note and its message.

Chat client: The mobile application will provide the user with an UI to create, send, receive and track text messages with the RC2 and other mobile devices. It uses the Extensible Messaging and Presence Protocol (XMPP) to provide instant messaging (text and voice messaging) functionality. The XMPP client interacts with an XMPP server that runs on the RC2.

Map client viewer: The aim of the map client viewer is to provide a view of the mobile application user's surroundings overlaid with relevant geospatial and mission-specific data as map layers.

Sensor manager: The sensor manager provides the map client viewer with access to the device sensor hardware, that is, cameras, GPS, gyroscopes and accelerometers. The sensor manager will provide methods to access the data from these devices using the Android SDK. Device's location data, provided by the GPS or GSM localization and images or videos captured by the mobile device, are geo-tagged and shared between the other C2I subsystems.

3.2.7. Exoskeleton controller

Figure 26 depicts the global software architecture of the exoskeleton component. This component is composed of the exoskeleton device associated with the haptic controller (HACO) running on a dedicated computer.

HACO is implemented on a Linux platform, running ROS and ROCK frameworks. ROCK is a software framework for the development of robotic systems. Running on top of the Orocos Real-Time Toolkit (RTT), it provides the tools to setup and run high-performance, real-time and reliable robotic systems (http://rock-robotics.org). It is used here to implement internal function of the exoskeleton or running in the haptic loop that requires real-time, deterministic and fast operations (red blocks) [31]. The haptic loop is typically running at 1 kHz. The other modules for configurations, communications with the RC2 and management of HACO that do not require high update rate are running in ROS (green blocks). The exchange of data between ROS and ROCK is performed through the ROCK/ROS bridge interface provided by the ROS framework. The following modules in **Figure 27** are implemented in HACO:

- HACO manager [ROS]:
  - Responsible for the configuration, management and monitoring of HACO
  - Interfaces the RC2 HMI manager through the command link that is a 'low-rate' communication link for remote status monitoring, commands and control parameters settings
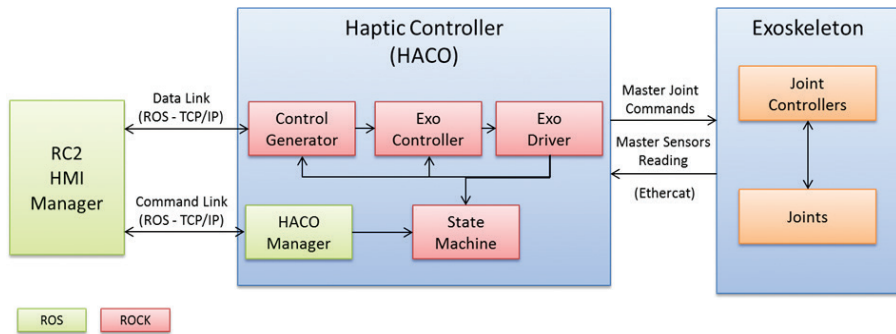

Figure 26. Global software architecture of the exoskeleton device (source: ICARUS).
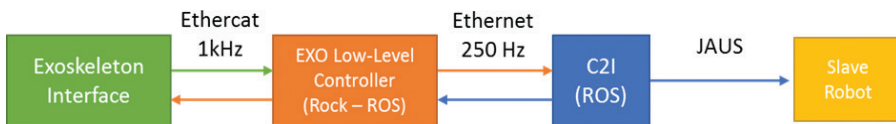

Figure 27. Exoskeleton control architecture (source: ICARUS).

- State machine [ROCK]:

  ○ Implements a state machine engine that allows defining HACO modules behaviour based on internal and external events. Internal events are events related to the operations of the exoskeleton (error in low-level joint controller's communication, exoskeleton switch triggering, etc.). External events are messages received from the command link (start/stop, control modes, etc.) and transmitted by the HACO manager.

- Control generator [ROCK]:

  ○ Interfaces the RC2 through the data link that is a 'high' rate communication link with the slave device (e.g. UGV arm) for haptic control exchanges. This link is used in both directions to receive position and forces from the slave side and also to send master (exoskeleton) position and force data to command the slave device.

  ○ Computes position or force feedback set points (Cartesian space) for the exoskeleton controller based on the inputs received from the slave and the current status of the exoskeleton.

  ○ Implements Cartesian Space features like guiding forces or Cartesian workspace limits.

- Exo Controller [ROCK]:

  ○ Computes the joint actuator commands according to the selected mode. This module is based on the knowledge of the exoskeleton kinematics and dynamics and is thus dedicated to this interface.

  ○ Converts the Cartesian set points provided by the control generator into joint set points for the exoskeleton (e.g. inverse kinematics, Jacobian transpose).

  ○ Implements the low-level haptic control schemes based on the comparison with the current exoskeleton sensor reading.

  ○ Implements joint space features like gravity compensation and software joint limits.

- Exo Driver [ROCK]:

  ○ Low-level interface with the joint controller boards embedded in the exoskeleton. The communication is based on EtherCAT that is well fitted for high-rate real-time and deterministic communication.

  ○ Sends master joint commands, reads exoskeleton sensors (position, torques and buttons) and publishes them for the other parts of the system (internal or external).

  ○ Implements the triggering system of the main haptic loop that is responsible to start at a constant rate (e.g. 1 kHz) one haptic loop step. The other blocks are driven by the output of the Exo Driver module.

Each joint of the exoskeleton is equipped with a joint controller that:

- Acquires torque and encoder signals

- Implements low-level control of the joint and PWM drive based on the received master joint commands (e.g. position or current set point)

- Interfaces the Exo Driver through EtherCAT communication bus

3.3. Portable hardware RC2 platform

Designed to operate in rough environment, the RC2 box has the full capability of controlling the UAVs, UGVs and USVs in both tele-operated and autonomous modes. It is equipped with a semi-rugged Dell E6430 ATG laptop docked on a rugged docking station, which is the interface between the robots and the user (**Figure 28**). Many options are available to control the drones: two embedded joysticks, a wireless game controller and a mouse. The user will also be able to monitor the different parameters of the mission thanks to an additional 15.6″ screen. Two powerful batteries give an operating time of 8 hours and power the different parts of the box: the laptop, the optional light, the fan, the screen and the powerful telescopic antenna. In order to communicate with the RC2, some external USB ports and Ethernet connector are also available. Easy to set up, the user will quickly be able to have the RC2 operational.

3.4. Exoskeleton hardware design and prototype

The force-feedback exoskeleton interface is composed of two main components, the 7 DOF arm (from the shoulder to the wrist) and the hand exoskeleton. Several modifications have been brought to the arm exoskeleton, compared to the first version built in the past for ESA

Figure 28. Portable RC2 CAD model (left) and finished RC2 rugged system (right) (source: ICARUS).

under the EXOSTATION project. The main modification is the material and manufacturing process used for the building of the structure. The new version is mainly based on rapid prototyping process (laser sintering) with alumide (composite aluminium and polyamide) and PA-GF (glass fibre-reinforced polyamide). Despite less rigidity of the manufacturing material, this allows a larger panel of shapes, as well as the integration of features (passing cable, fixation holes, etc.). Finite Element Analysis (FEM) analysis allows us to design a structure with comparable mechanical behaviour than the first version, with a slight reduction of weight. The kinematic configuration of the shoulder has also been updated in order to increase the achievable workspace within the exoskeleton, mainly when the arm is in the vicinity of the body. A half-circle curved guiding rail replaces now the full circle bearing on the upper arm. That improves the mechanical interaction with the body as well as facilitates the installation inside the exoskeleton.

The large unmanned ground vehicle is equipped with a 5DOF manipulator arm (**Figure 29**). The manipulator is hydraulic powered and consists of three rotational joints and two hydraulic cylinders. All five joints are feedback controlled by two external FPGA-based low-level controllers. These allow the actuation of the manipulator from remote and in an automated way. For each of the feedback controlled actuators, it is possible to set a desired position and a desired velocity and to receive the actual sensor values for the actuator positions and velocities. Additionally, the actual pressure values in the hydraulic joints are provided. The controllers are interfaced by the computer which runs the main control software of the Large Unmanned Ground Vehicle (LUGV). There the joint positions and velocities are transformed to a more convenient and sophisticated interface. All joint actuator sensor and control values are converted to joint angles and angular velocities which meet the Denavit-Hartenberg convention. The high-level control software is also responsible for safe operation and initialization of the two low-level controllers. Therefore, the operational state of both controllers is observed and synchronized, and the validity of the inputs is checked. This avoids unexpected behaviour during the initialization and operation phase, e.g. sudden movements or malfunction of single manipulator joints.

Figure 29. (i) SAM exoskeleton upper part advanced design and rapid prototyping part integration test. (ii) LUGV with extended manipulator (source: ICARUS).

4. C2I subsystem integration and field deployment

4.1. Map interface

The central widget in **Figures 30** and **31** of the RC2 is the map interface.

- Multiple layers are provided as base maps, mission planning and robot positons.

- A zoom and pan option is provided for the user to navigate through the map layers using a standard mouse interface.

- The base maps consist of layers for

    ○ Military maps (e.g., test site Marche-en-Famenne)

    ○ Satellite, elevation and vectorial (roads, buildings, etc.) maps for the Moia CTC test area, Spain

    ○ Satellite and vectorial (roads, buildings, etc.) maps for the Portugal CINAV naval base

- Operational and mission planning layers consist of:

    ○ Robot layer

    ○ Waypoint layer

    ○ Sector layer
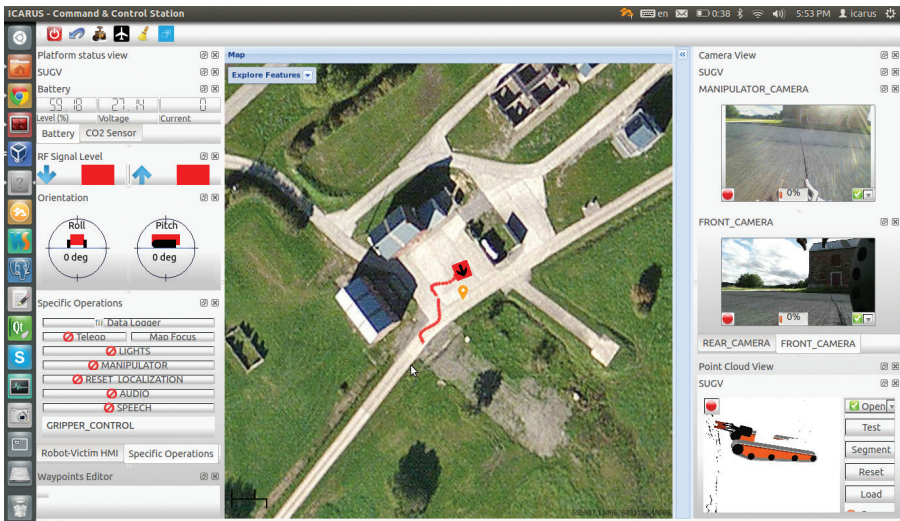
Figure 30. C2I interface with maps, sensor visualizations and robot control (SUGV) (source: ICARUS).
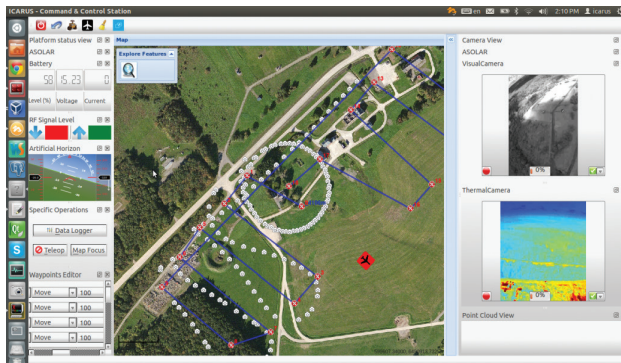
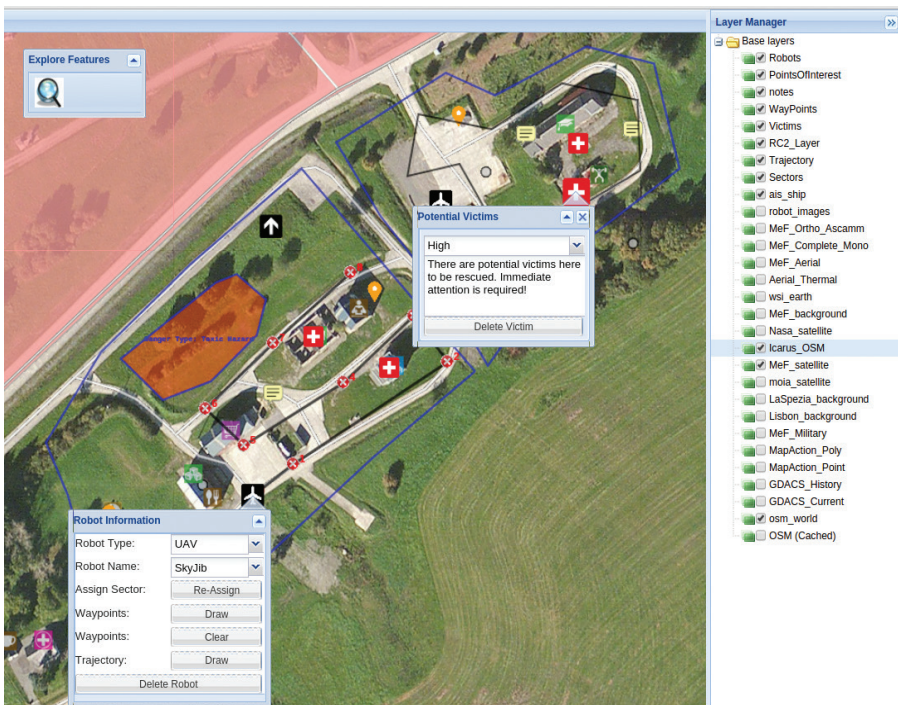**Figure 31.** C2I Mission plan execution with AtlantikSolar UAV (source: ICARUS).

Figure 32. Mission planning interface in the MPCS (source: ICARUS).

4.2. Mission planning and operation

At the MPCS, the mission authoring tool illustrated in **Figure 32** consists of the following:

- Adding virtual robots to the map at desired locations and constraining their activity within sectors.

- A sector can be freely drawn on the map using the 'map context menu->draw sector' tool. The sector drawing tool uses consecutive clicks on the map from the user to draw the polygon. The sector polygon can be modified by selecting the sector and option to drag and resize the sector and also deleted.

- A robot within the sector is then selected by the user and the associated context menu on the map allows the user to annotate the map with a set of waypoints associated with the robot.

- Each waypoint has an associated entry in the waypoint editor where the user can set specific parameters such as waypoint type (start, loiter, stop), velocity, altitude, waypoint tolerance, path tolerance, etc.

- On selecting the robot, a popup menu is displayed indicating user-driven interactions with the robot such as sending waypoints to the planner or the robot, hiding or showing waypoints on the map, constraining the robot to its bounding sector, etc.

4.3. Automated mission planner

For the automated mission planner at the MPCS, the following requests are served (**Figure 33**):

- **Path planning**: The algorithm used is CUDA-based implementation of wavefront. The algorithm works with a 2D occupation grid map with user-defined waypoints as inputs (**Figure 33**), generated based on the semantic representation of the environment.

- **Global path planning**: The planners are able to give an answer to the travelling salesman problem. The implementation is based on a hill climbing algorithm, which allows for finding locally optimal solutions like scanning a sector as seen in **Figure 33**.

- **Find optimal observation point**: The planners are able to answer the question of optimal observation point of requested object with a given set of sensor. The representation of the environment is generated from the semantic model (**Figure 34**).

- **Find optimal repeater position**: Functionality for finding a spot from which the UGV could be working as a repeater. The query takes two disconnected signal sources that are weak to connect directly and simulates the disruption of the signal in the environment (**Figure 35**).

The mission planners are using supporting tools. The most important one is the semantic environment model generation tools. The tools take 3D point clouds of a given area and generate a semantic representation of given area based on them. A simple model may be also generated based on GIS information. The semantic map divides the points into three main categories: ground, structured and unstructured (**Figure 36**). This allows for segmentation of single objects and making decision about traversability of a given terrain. **Figure 37** shows the traversability analysis. Green points are traversable while red are not. Three examples in
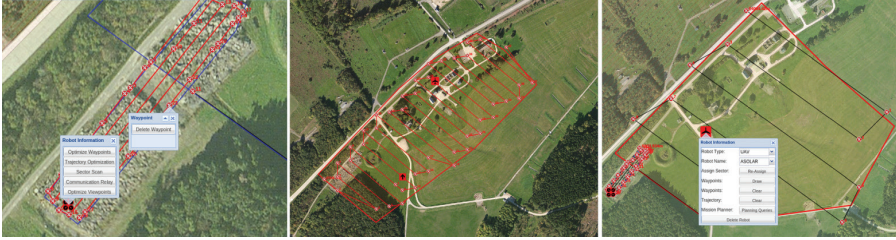
Figure 33. (i) Automated mission planning queries. (ii) Sector scan query. (iii) Optimization of waypoint query (source: ICARUS).
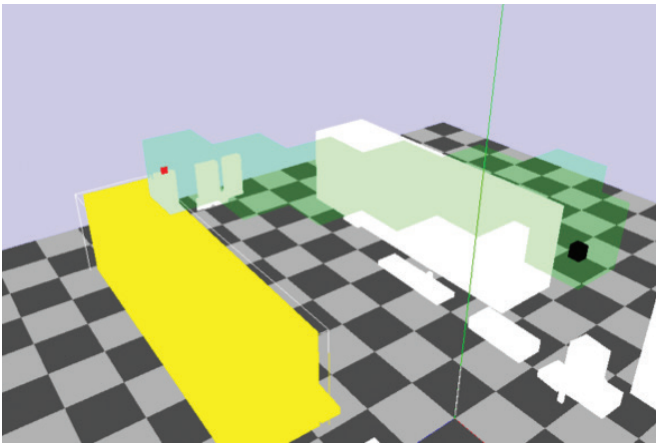

Figure 34. Object observation point query visualization: black box, robot pose; red box, new robot pose (source: ICARUS).

the picture were generated using different robot models. The semantic model may be used to generate virtual model of the terrain.

4.4. RC2 visualization and control

Sensor visualizations in **Figure 38** include the following dockable widgets:

- **Robot pose**:
  - ○ The global NSEW orientation of the robot is shown on the map with the robot icon indicating the heading with an arrow.
  - ○ The UAVs are provided with an artificial horizon that shows the roll and pitch, altitude and the rate of climb.
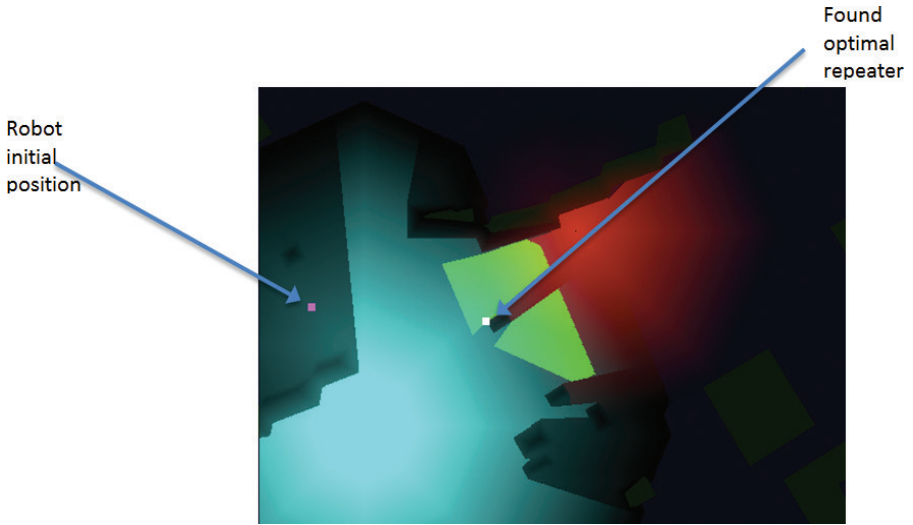 - ○ UGVs have two independent indicators for the roll and pitch of the robot.

Figure 35. Robot as repeater query: red, range of the first communication source; blue, range of the second communication source; green, potential positions that allow work as repeater (source: ICARUS).
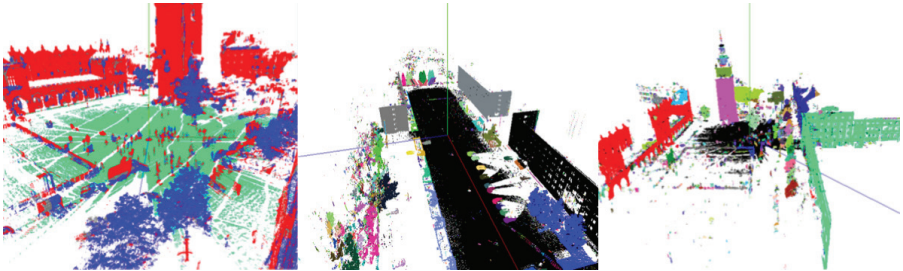

Figure 36. (i) Point cloud classification of data from geodetic scanner. (ii and iii) Scene segmentation examples (source: ICARUS).
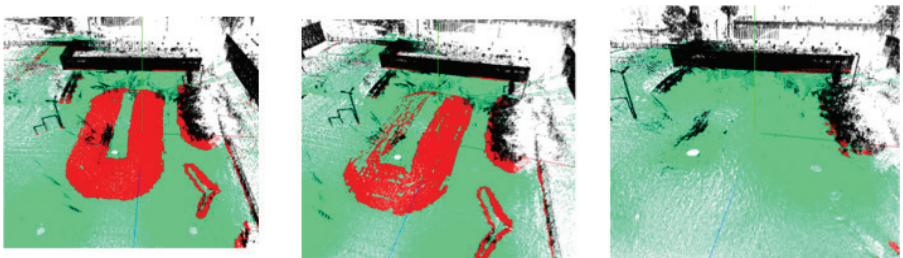

Figure 37. The traversability analysis: traversability for UGV with 10° max slope, 18° max slope and 44° max slope (source: ICARUS).
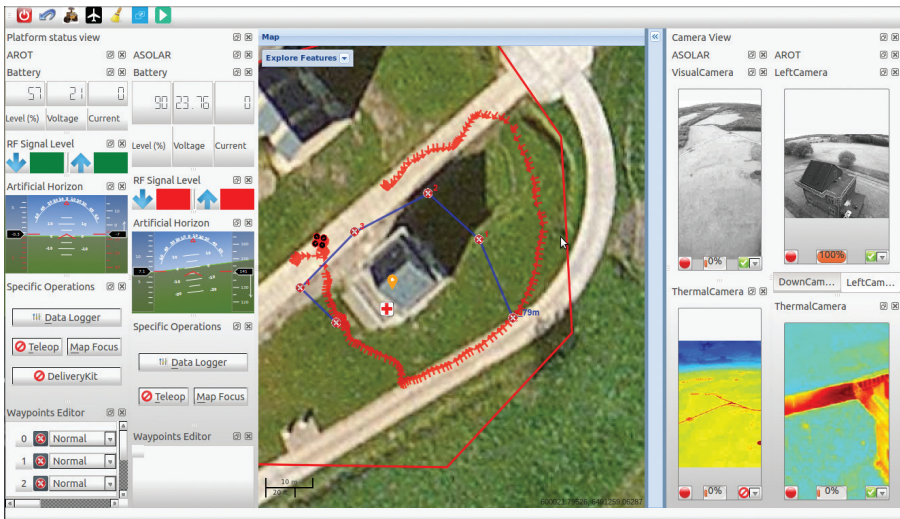
Figure 38. C2I with AtlantikSolar UAV and AROT quadrotor (source: ICARUS).

- **Camera viewer**:
  - ○ This component renders all the cameras that are streaming videos from a robot.
  - ○ It contains dockable windows that can be resized, tabbed or undocked from the parent window to be positioned anywhere by the user. The rendered video resizes to the window while maintaining its aspect ratio.

- **Waypoint editor**:
  - ○ Each waypoint associated with a robot is displayed in a list form.
  - ○ Every parameter of the waypoint can be edited from this editor such as waypoint type (start, loiter, stop), velocity, altitude, waypoint tolerance, path tolerance, etc.

- **Joystick selector**:
  - ○ This is a single button to switch the control of a robot to tele-op mode to select the appropriate joystick control.

- **Point cloud renderer**:
  - ○ This widget can render raw point clouds from Lidar sensors or the global 3D map of the scanned that are provided by the robot.

- **Battery and wireless status**:
  - ○ These are two independent-level indicators showing the current energy levels of a robot and the quality of the wireless network link (in percentage).

A PS3 game pad connected to the RC2 via Bluetooth has been configured and interfaced with the C2I to tele-operate a robot. There are currently four axes of control and multiple buttons which can be used according to the type of platform. The joystick was used to control the UGVs and the quadrotors. Tele-operation of virtual robots in simulators has also been implemented and tested.

4.5. RC2-integrated training with simulators

The RC2 has been integrated with two simulators as per the reference network architecture in **Figure 39** for training purposes over ROS:

- The USAR training simulator (**Figure 40**) is capable of streaming virtual data such as videos from multiple virtual cameras, virtual global positon and orientation of the robot. This data can be rendered in the C2I similar to that of a real robot. Tele-operation of the virtual robot is also possible using the PS3 joystick controller. Remote streaming and control of the robot were achieved over the Internet with the C2I operating in Brussels and the UGV simulator hosted on a server in Poland within a VPN with standard (expected) delays over the Internet.
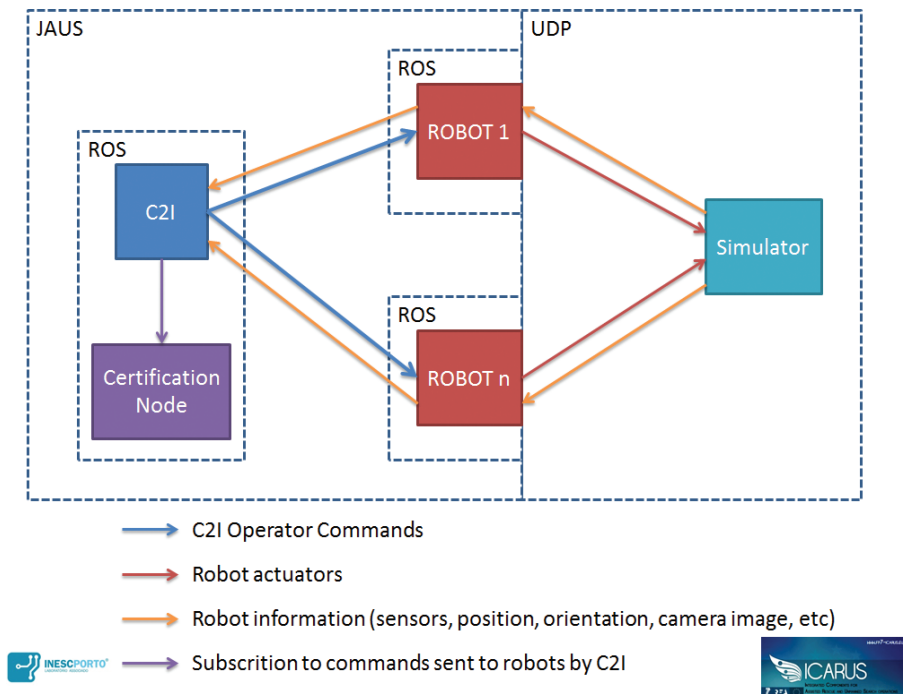

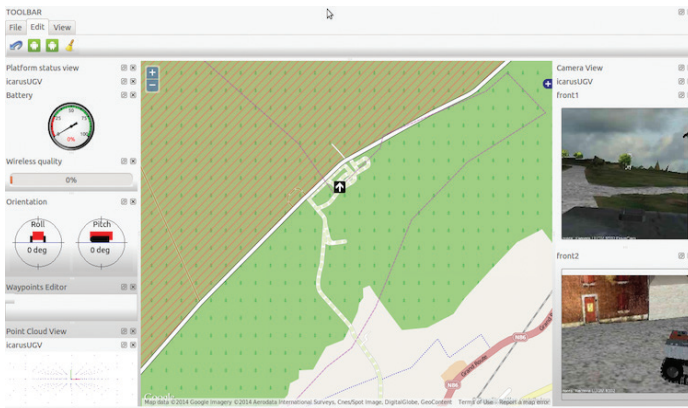**Figure 39.** Maritime simulator network architecture (source: ICARUS).

Figure 40. RC2 with feed from ground robots in simulated environment (source: ICARUS).

- The MSAR simulator (**Figure 41**) provides virtual data such as videos from multiple virtual cameras, virtual global positon and orientation of the robot, battery level and wireless link quality. These sensor data can be visualized in the C2I similar to a real USV.

Since the simulator will only simulate the sensorial/physical aspects of the robots, the connection between the C2I and the simulation is transparent and does not require any extra overhead in integration. The figure below shows the final integration between the simulator and the C2I.

4.6. C2I-JAUS capabilities

The ICARUS interoperability standard JAUS has been integrated with the C2I. The 'JAUS-fleet' is responsible for the automatic discovery of a robot within the JAUS network environment. The 'JAUS-fleet' sends a ROS-robot profile message indicating the addition of a new robot to the network. The C2I responds to this dynamic discovery by configuring the front-end user interface and visualizations corresponding to the type of robot (UAV, UGV or USV). Sensor data from the robot and commands from the C2I to the robot are sent via ROS topics which are also dynamically generated. The current level of compatibility of the C2I through the JAUS interface is as follows:

- Multiple-camera video streaming

- Four axis Joystick commands

- Sending waypoints with metadata (path and waypoint tolerance) to the robot

- Global pose of the robot (GPS and inertial data)

- Dynamic robot platform discovery
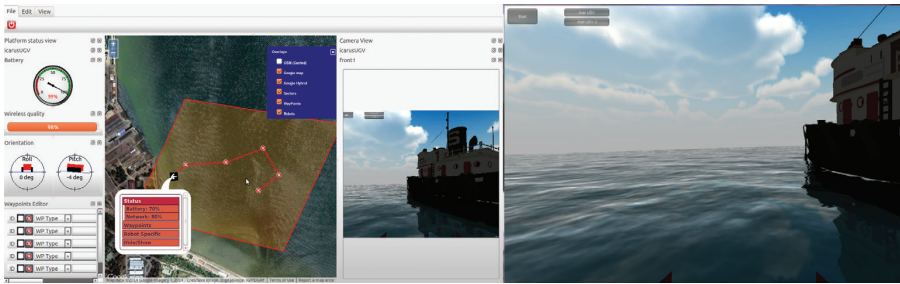
- Multi-robot operation capability

Figure 41. Integration result (left, C2I; middle, robot controller; right, maritime simulator) (source: ICARUS).

4.7. GIS datasets

4.7.1. Maps and data

The following environmental data have been integrated into ICARUS system:

- Moia-BBOX (41.818728 2.1773529, 41.803886 2.1482563) (**Figure 42**):

  ○ Vectorial data obtained from open street maps (OSM). Deployed in GeoServer and MapServer

  ○ Orthoimages for the Moia region and surroundings, obtained from the Spanish Geographical Institute (IGN). Deployed in MapServer

  ○ Vectorial data depicting slope, altitude, hydrography and roads. Deployed in MapServer and GeoServer

  ○ Vectorial data for Catalonia villages, boundaries, regions, municipalities and provinces. Deployed in MapServer and GeoServer

- Marche-en-Famenne-BBOX (50.264326 5.3996086, 50.254010 5.3782368) (**Figure 43**):

  ○ Vectorial data obtained from open street maps (OSM). Deployed in MapServer and GeoServer

  ○ Pyramidal raster data for the Marche-en-Famenne region, obtained from the Royal Military Academy (RMA). Deployed in GeoServer

  ○ Top-view raster from the test area, obtained from the Royal Military Academy (RMA). Deployed in GeoServer

- Lisbon-BBOX (38.667258 -9.100424, 38.649694 -9.1492938) (**Figure 44**):

  ○ Vectorial data obtained from open street maps (OSM). Deployed in GeoServer and MapServer

  ○ Raster satellite maps from openly available sources such as NASA earth, local government agencies, ESA Copernicus satellite imagery etc.
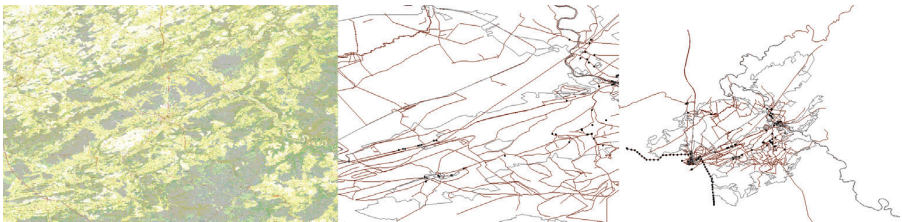
Figure 42. Moia map data (source: ICARUS).

Figure 43. Marche-en-Famenne map data (source: ICARUS).

Figure 44. Lisbon map data (source: ICARUS).

Apart from environment GIS information, ICARUS database schema defines some geospatial entities that have been defined and published as layers (zones, sectors, victim status, trajectories, structures, robots, missions, mission features, GDACS items, floor plans and waypoints)—GeoServer, MapServer.

4.7.2. External data services

GDACS provides a RSS with worldwide disaster event information. This data source has current disaster information and related information such as images, documents, URLs, etc.

We have developed an application that dumps information to a spatial/GIS database. The data have the geographic positions that allow us to depict them on a map and consult related information. In order to safeguard the records related to each item of disaster, file types that are of interest are also copied. Item information retrieval in a Popup with right button click shows historic disasters evolution (**Figure 45**).

4.8. Data fusion module

This module is currently divided into two big objectives: map generation and map segmentation. **Figures 46** and **47** show the result of both entities, respectively.

4.9. Mobile application for first responders

The mobile application user interface (**Figure 48**) has been deployed on Android platform running version 4.2.2 or later. The application will provide the following features:

- Maps: The application connects to the MPCS and RC2 map server interfaces using HTTP and downloads map layers and associated content from the GIS. The map view will in addition to the maps overlay information such as current position of the user, team and robot positions.

- Text and image notes: The application provides a note-taking tool for the user to create text, image and video notes and tag them to his current position on the map.

- Other map features include the position of victims, points of interest, sector of operations, multiple base map layers (OSM, satellite, military maps, etc.) and a simple instant messaging platform for text communication between RC2 operators and other mobile devices.

4.10. Exoskeleton interface with UGV manipulator

The exoskeleton was employed with the C2I to provide an intuitive manipulation interface for manipulators of the sensor visualizations and robot control (SUGV) and the LUGV. During operation, the operator was wearing the exoskeleton device beside the C2I system in order to be able to see the on-board slave robot's cameras (e.g. zoom on the gripper) and the slave robot arm model simulations (view of robot state based on collected data), helping him for precise manipulation and operations. Thanks to the triggering system, it was easy to enable and disable
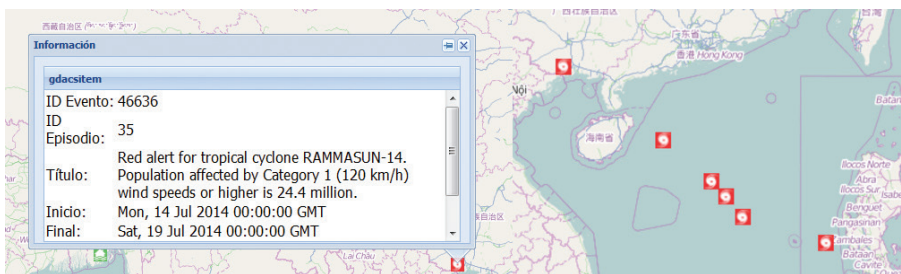

Figure 45. Inspection of GDACS information in the C2I map application (source: ICARUS).
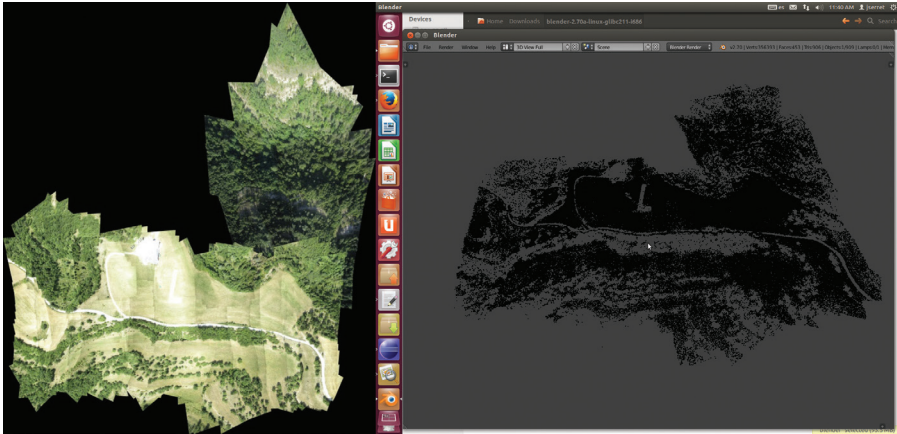
Figure 46. Fast mapping results in 2D (textured, left) and 3D (sparse cloud, right) for flights in Moià (source: ICARUS).
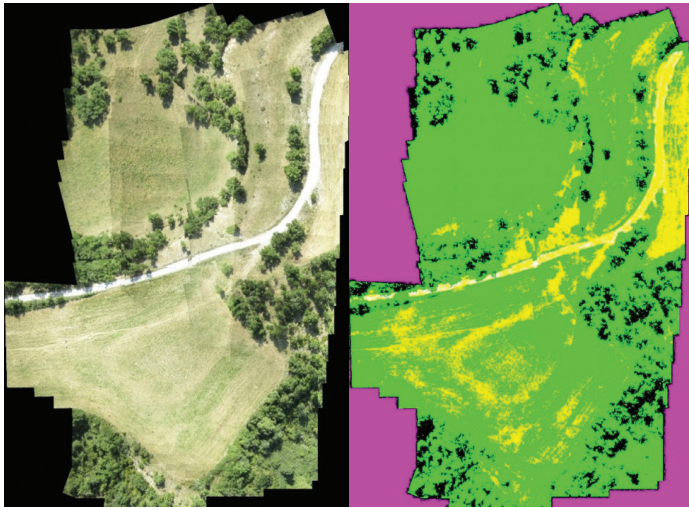
Figure 47. (Left) Original map used to be classified, the same as shown in ground truth selection and preview of results over a map generated during the trials. (Right) Segmented map, prediction done by the service implemented in ROS (source: ICARUS).

the control link with the slave arm. **Figure 46** illustrates the operation of the SUGV with the exoskeleton during the final demo. The exoskeleton was used to control with dexterity the slave arm with the objective to open a door handle. Compared to a standard joystick or pad control-ler, this solution allowed being more accurate and quicker, with the capacity to transfer to the robot the good motion for the handle operation. **Figures 49** and **50** highlight the operation of the LUGV with the exoskeleton that was performed during the preparation phases.

Figure 48. User interface of the mobile application (source: ICARUS).

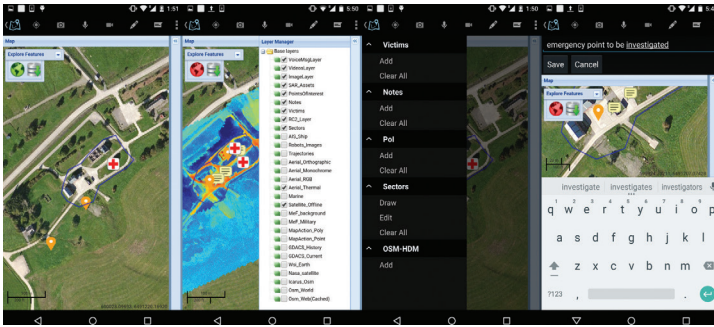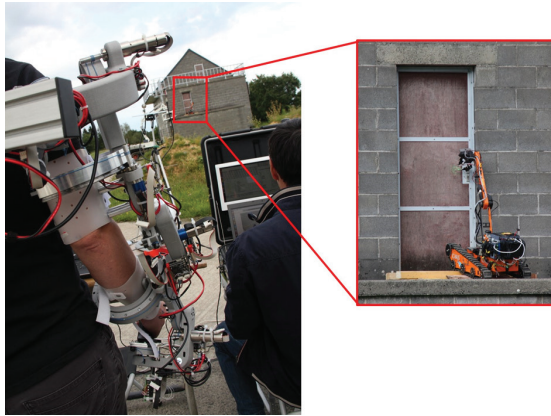**Figure 49.** Control of the SUGV with the exoskeleton interface in operational scenario to operate a door handle (source: ICARUS).

Figure 50. Control of the LUGV with the exoskeleton interface to grab objects (source: ICARUS).

5. Conclusions

The C2I system of the ICARUS project is an essential set of hardware and software components, instrumental in providing interfaces for SAR responders to get a common operation picture for supervising SAR tasks. The MPCS, RC2, exoskeleton and mobile field devices of the C2I system provide a distributed capability for planning and controlling unmanned robots and SAR personnel, thus improving the effectiveness of the response to crisis situations. Offline mission planning capability coupled with human in the loop commanding a fleet of tele-operated and semi-autonomous robots during SAR operations demonstrated the effectiveness of such a system. Future enhancements to the C2I include runtime operational mission planning and immersive 3D HMI interfaced with advancements in robot autonomy and fault-tolerant multi-robot cooperation [32]. Field demonstrations of the C2I system with SAR personnel assisted by unmanned systems provide an outlook for implementing such systems into mainstream SAR operations in the future. The flexibility of integrating the C2I with diverse robotic platforms will enable a large variety of robots to be tested, evaluated and eventually used in SAR operations.

Author details

Shashank Govindaraj[1]*, Pierre Letier[1], Keshav Chintamani[1], Jeremi Gancet[1], Mario Nunez Jimenez[2], Miguel Ángel Esbrí[2], Pawel Musialik[3], Janusz Bedkowski[3], Irune Badiola[4], Ricardo Gonçalves[5], António Coelho[5], Daniel Serrano[6], Massimo Tosa[7], Thomas Pfister[7] and Jose Manuel Sanchez[8]

*Address all correspondence to: shashank.govindaraj@spaceapplications.com

1 Space Applications Services NV, Zaventem, Belgium

2 ATOS, C/Albarracín, Madrid, Spain

3 Institute of Mathematical Machines, ul. Krzywickiego, Warsaw, Poland

4 Estudios GIS, Parque Tecnológico de Álava – Edificio E7 C/Albert Einstein, Miñano (Álava), Spain

5 INESC TEC – Instituto de Engenharia de Sistemas e Computadores, Tecnologia e Ciência and Faculdade de Engenharia da Universidade do Porto, Campus da FEUP, Rua Dr. Roberto Frias, Porto, Portugal

6 Eurecat, Av. Universitat Autònoma, Cerdanyola del Vallès, Spain

7 Technische Univeristät Kaiserslautern, Gottlieb-Daimler-Strasse – Geb., Kaiserslautern, Germany

8 IntegraSys SA, Calle Esquilo, Madrid, Spain

References

[1] Murphy RR, Peschel J. On the human-computer interaction of unmanned aerial system mission specialists. IEEE Transactions on Human-Machine Systems. 2013;**43**:53-62

[2] Kruijff GM, Kruijff-Korbayová I, Keshavdas S, Larochelle B, Janíček M, Colas F, Liu M, Pomerleau F, Siegwart R, Neerincx MA, Looije R, Smets NJJM, Mioch T, van Diggelen J, Pirri F, Gianni M, Ferri F, Menna M, Worst R, Linder T, Tretyakov V, Surmann H, Svoboda T, Reinštein M, Zimmermann K, Petříček T, Hlaváč V. Designing, developing and deploying systems to support human-robot teams in disaster response. Advanced Robotics, Special Issue on Disaster Response Robotics. 2014;**28**(23):1547-1570.

[3] Gancet J, Motard E, Naghsh A, Roast C, Arancon MM, Marques L. User interfaces for human robot interactions with a swarm of robots in support to firefighters. In: IEEE International Conference on Robotics and Automation (ICRA); 3-7 May 2010; IEEE; 2010. DOI: 10.1109/ROBOT.2010.5509890

[4] Doroftei D, De Cubber G, Chintamani K. Towards collaborative human and robotic rescue workers. In: 5th International Workshop on Human-Friendly Robotics (HFR2012); 18-19 October; Brussels, Belgium. 2012

[5] Govindaraj S, Chintamani K, Gancet J, Letier P, Van Lierde B, Nevatia Y, De Cubber G, Serrano D, Bedkowski J, Armbrust C, Sanchez J, Coelho A, Palomares ME, Orbe I. The ICARUS project – Command, control and intelligence (C2I). In: Safety, Security and Rescue Robots; October 2013; Sweden: IEEE; 2013

[6] UAV Factory Portable Ground Control Station. Available from: http://www.uavfactory.com/product/16

[7] OpenPilot Mission Planner. Available from: http://wiki.openpilot.org/display/Doc/Open-Pilot+Documentation

[8] Maza I, Ollero A, Casado E, Scarlatti D. Classification of multi-{UAV} architectures. In: Handbook of Unmanned Aerial Vehicles. Netherlands: Springer; 2014; pp. 953-975.

[9] QGroundControl Ground Control Software. Available from: http://www.qgroundcontrol.org/

[10] Gancet J, et al. DexROV: Dexterous undersea inspection and maintenance in presence of communication latencies. 4th IFAC Workshop on Navigation, Guidance and Control of Underwater Vehicles NGCUV, 2015.

[11] Zunaid Kazi , Marcos Salganicoff , Matthew Beitler , Shoupu Chen , Daniel Chester , Richard Foulds. Multimodal User Supervised Interface and Intelligent Control (MUSIIC) for Assistive Robots. AAAI FALL SYMPOSIUM SERIES ON EMBODIED LANGUAGE AND ACTIO, MIT, 1995.

[12] Vicente KJ, Rasmussen J. Ecological interface design: Theoretical foundations. IEEE Transactions on Systems, Man, and Cybernetics, 1992;**22**(4):589-605.

[13] Vicente KJ. Ecological interface design: A research overview. Paper presented at the Analysis, Design and Evaluation of Man-Machine Systems 1995, the 6th IFAC/IFIP/IFORS/IEA Symposium, Cambridge, MA; 1995.

[14] Burns CM, Hajdukiewicz JR. Ecological Interface Design. Boca Raton, FL: CRC Press; 2004

[15] Woods DD. Toward a theoretical base for representation design in the computer medium: Ecological perception and aiding human cognition. In: Flach J, Hancock P, Caird J, Vicente K, editors. Global Perspectives on the Ecology of Human-Machine Systems. Vol. 1. Hillsdale, NJ: Lawrence Erlbaum Associates, Inc. Publishers; 1995; pp. 157-188

[16] Chen JYC, Barnes MJ, Harper-Sciarini M. Supervisory control of multiple robots: Human-performance issues and user-interface design. IEEE Transactions on Systems, Man, and Cybernetics – Part C: Applications and Reviews, July 2011;**41**(4)

[17] Thomas LC, Wickens CD. Effects of Display Frames of Reference on Spatial Judgments and Change Detection, Technical Report ARL-00-14/FED-LAB-00-4, September 2000

[18] De Cubber G, Doroftei D, Serrano D, Chintamani K, Sabino R, Ourevitch S. The eu ICARUS project: Developing assistive robotic tools for search and rescue operations. In: 2013 IEEE International Symposium on Safety, Security, and Rescue Robotics (SSRR). 2013; 1-4 pages.

[19] Lewis M. Human interaction with multiple remote robots. In: Kaber D, editor. HF Reviews on Human Performance in Teleoperation and Beyond. Vol. 9. HFES, 2013; pp. 131-174

[20] Gerkey BP, Matarić MJ. A formal analysis and taxonomy of task allocation in multi-robot systems. International Journal of Robotics Research. 2004;**23**(9):939-954

[21] Kevin B. Bennett, John M. Flach. Display and Interface Design: Subtle Science, Exact Art. March 9, 2011 by CRC Press, ISBN 9781420064384

[22] Chrobocinski P, Zotos N, Makri E, Stergiopoulos C, Bogdos G. DARIUS project: Deployable SAR integrated chain with unmanned systems. In: 2012 International Conference on Telecommunications and Multimedia (TEMU), 30 July –1 August 2012; pp. 220, 226

[23] Yan Z, Jouandeau N, Cherif AA. A survey and analysis of multi-robot coordination. International Journal of Advanced Robotic Systems, 2013;**10**(1).

[24] Ingrand F, Ghallab M. Robotics and artificial intelligence: A perspective on deliberation functions. AI Communications, 2014;**27**(1):63-80. Available from: http://dl.acm.org/citation.cfm?id=2594611.2594619

[25] Korsah GA, Stentz A, Dias MB. A comprehensive taxonomy for multi-robot task allocation. The International Journal of Robotics Research. 2013;**32**(12):1495-1512. DOI: http://doi.org/10.1177/0278364913496484

[26] Kostavelis I, Gasteratos A. Semantic mapping for mobile robotics tasks: A survey. Robotics and Autonomous Systems. 2015;**66**:86-103

[27] Ricks B, Nielsen CW, Goodrich MA. Ecological displays for robot interaction: A new perspective. Proceeding of the IEEE IROS, Sendai, Japan. 2004;384-404

[28] Nielsen CW, Goodrich MA, Ricks B. Ecological interfaces for improving mobile robot teleoperation. IEEE Transactions on Robotics and Automation. 2007;**23**:927-941.

[29] Balta H, Bedkowski J, Govindaraj S, Majek K, Musialik P, Serrano D, Alexis K, Siegwart R, De Cubber G. Integrated data management for a fleet of search-and-rescue robots. Journal of Field Robotics. 2016. DOI: 10.1002/rob.21651

[30] Damilano L, Guglieri G, Quagliotti F, Sale I, Lunghi A. Ground control station embedded mission planning for UAS. Journal of Intelligent & Robotic Systems. 2013;**69**(1-4):241-256

[31] Letier P, Motard E, Ilzkovitz M, Preumont A, Verschueren JP. SAM: Portable haptic arm exoskeleton upgrade technologies and new applications fields. In: Proceeding of the 11th ESA Workshop on Advanced Space Technologies for Robotics and Automation, Noordwijk, April 2011.

[32] Parker LE. {ALLIANCE}: An architecture for fault-tolerant multi-robot cooperation. IEEE Transactions on Robotics and Automation, 1998;**14**(2):220-240

ICARUS Training and Support System

Janusz Będkowski, Karol Majek, Michal Pełka,
Andrzej Masłowski, Antonio Coelho,
Ricardo Goncalves, Ricardo Baptista and
Jose Manuel Sanchez

Additional information is available at the end of the chapter

Abstract

The ICARUS unmanned tools act as gatherers, which acquire enormous amount of information. The management of all these data requires the careful consideration of an intelligent support system. This chapter discusses the High-Performance Computing (HPC) support tools, which were developed for rapid 3D data extraction, combination, fusion, segmentation, classification and rendering. These support tools were seamlessly connected to a training framework. Indeed, training is a key in the world of search and rescue. Search and rescue workers will never use tools on the field for which they have not been extensively trained beforehand. For this reason, a comprehensive serious gaming training framework was developed, supporting all ICARUS unmanned vehicles in realistic 3D-simulated (based on inputs from the support system) and real environments.

Keywords: training systems, support systems, real-time 3D reconstruction

1. Introduction

The ICARUS Training and Support system provides the command and control component that integrates different data sources of spatial information, such as maps of the affected area, satellite images and sensor data coming from GIS database and the unmanned robots in order to provide a situation snapshot to the rescue team that makes the necessary decisions. The system is implemented based on the concept of High Preformance Computing (HPC) in the Cloud . The integration and visualization of maps derived from UxVs is the main functionality of the proposed HPC solution. These maps are available in the Cloud; therefore, the information concerning disaster can be distributed over Ethernet within the connectivity constraints.

The second important functionality is the ICARUS serious games in the Cloud which are used for end-user training purposes. The proposed HPC solution is capable of streaming serious games over Ethernet; therefore, ICARUS international teams can train simultaneously from offices in different countries. The third functionality is the integration-evaluation cycle using HPC. ICARUS partners had access to the server installed in the Data Centre, which means that the integration between the different teams could be performed using this tool during project development and evaluation phase.

2. State of the art

The main and most important task of rescue services during a major crisis is to search for human survivors on the incident site. As such an endeavour is complex and dangerous, it often leads to loss of lives among the rescuers themselves. It is evident that the Unmanned Search And Rescue devices can improve search and rescue process. Many research efforts towards the development of unmanned SAR tools have been made [1]. One of these efforts is Neptus, a C3I (Command, Control, Communication and Information) framework, which aims to support coordinated operation of heterogeneous teams, including several types of UVs and human beings [2]. Another example is the German project I-LOV, which establishes a framework for integration of mobile platforms into a victim search mission [3]. Numerous attempts to use robotic systems in crisis situations were made: the 2001 World Trade Center attack [4], the 2004 earthquake in Mid Niigata, the 2005 USA hurricanes [5] or the 2011 Japan tsunami. Papers [3] and [6] give a broad overview of the effort done in this area. This research effort stands in contrast to the practical reality in the field, where unmanned SAR tools have great difficulty finding their way to the end users, due to a number of remaining bottlenecks in the practical applicability of unmanned tools [7].

The Training and Support system concerns the Serious Games (SGs) that are becoming increasingly popular in the corporate and research communities. However, there are still different definitions of what a serious game is. In this chapter, SGs are defined as follows: serious game applications which take advantage of all the features that make games fun and engaging and use them to empower training [12], promoting the trainees' interest by making the educational subject more exciting. Training is defined as "an organized activity aimed at imparting information and/or institutions to improve the recipient's performance or to help him or her attain a required level of knowledge or skill" [13]. Using games and structured learning activities in training is an excellent way to bring key topic areas to the learner [14]. On the SG taxonomy defined by Sawyer and Smith [15], games for training fall in areas like government, defence, education and industry. They cover different aspects such as occupational safety, skills, communications and orientation (e.g. Ref. [16]).

Currently, several tools exist to simulate and visualize unmanned vehicles operating since this type of software allows overall cost reduction and provides a platform for safer and faster testing. The relevant example is the USARSim [17], an open source framework built on top of

Unreal Engine [18] to simulate multirobots and environments. In addition, interfacing with the Mobility Open Architecture Simulation and Tools framework (MOAST [19]) provides a modular control system of robot control and customization. It allows the system users to add their own modules or alter the existing ones in order to obtain more complex robots than USARSim is capable of implementing. Another example of such a tool is Webots [20], which also allows some degree of robot customization at the shape and attribute level. This tool also has the advantage of allowing robots to be independent from the tool and communicate with it remotely through TCP/IP. To handle physics simulation, Webots uses Open Dynamic Engine (ODE [21]). Recent most popular simulation tool for multirobot systems is Gazebo integrated with ROS (Robot Operating System) [22].

The ICARUS (Integrated Components for Assisted Rescue and Unmanned Search Operations) project is a large European research project with 24 partners from 10 European countries. It concentrates on the development of unmanned surface vehicles (USV), unmanned air vehicles (UAV) and unmanned ground vehicles (UGV) and search and rescue technologies for detecting, locating and rescuing humans. All these systems need to share information between them, and interoperability is an important issue to take in account [8, 9].

3. Hardware

The proposed HPC solution for the ICARUS project is based on the Server Supermicro RTG-RZ-1240I-NVK2 shown in **Figure 1**. This server provides the NVIDIA GRID Technology. NVIDIA GRID technology includes NVIDIA CUDA architecture, which means that the server provides a parallel programming solution for processing large data sets. These advantages of the server make it relevant for ICARUS needs, as data from many sources (UxV) can be efficiently integrated and visualized over Ethernet.

3.1. Key features

1. Dual socket R (LGA 2011) supports Intel® Xeon® processor E5-2600 v2 family

2. Up to 512GB ECC DDR3, up to 1866MHz; 8x DIMM sockets

3. 3x PCI-E 3.0 x16 slots (supports GPU cards), 1x PCI-E 3.0 x8 (in x16) low-profile slot

4. Integrated IPMI 2.0 with KVM and Dedicated LAN

5. Intel® X540 10GBase-T Controller

6. 4x Hot-swap 2.5″ SATA3 Drive Bays

7. 1800W Redundant Power Supplies Platinum Level (94%+)

8. 10x Counter rotating fans w/optimal fan speed control

9. Smart server management tools.

Figure 1. The HPC solution for the ICARUS project is based on the Server Supermicro RTG-RZ-1240I-NVK2 (source: ICARUS).

3.2. Server's specification

- 2x Intel 2.8GHz Ten-Core Xeon CPU's (Max Turbo 3.6GHz)

- 256GB (16x16GB) Supermicro DDR3 Registered ECC Memory

- 2x 240GB Enterprise Class SATA 2.5" SSD's extended by 1TB SSD Samsung Hard Drive

- 2xNVIDIA GRID K2 cards (4GPUs in Total).

3.3. Ruggedized chassis

Meant to be deployed in tough environmental conditions, the server was embedded into a ruggedized chassis as shown in **Figure 2**. The chassis can easily be carried by two people. It is even possible for a single person to carry the chassis. It protects the server from vibrations and mechanical stress. **Figure 3** shows the final fully equipped HPC solution for the ICARUS
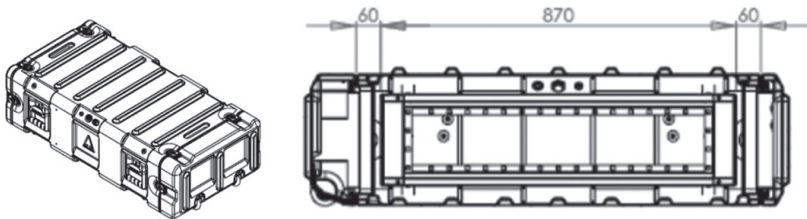

Figure 2. Chassis diagram and measurements (source: ICARUS).

Figure 3. Fully equipped HPC solution for the ICARUS project (source: ICARUS).

project. It is extended by a mobile display-keyboard-mouse component for on-site server management purpose. It is a fully integrated and autonomous solution. It requires 2kV AC power. The communication system is based on a WiFi router that can establish a local network within a range of 25 m. If Ethernet access is available, the server is connected directly to the network.

4. Software infrastructure

The Training and Support system in the Cloud is designed based on two models: VDI (Virtual Desktop Infrastructure, **Figure 4**) and SaaS (Software as a Service, **Figure 5**). These models support vGPU (virtualization of Graphic Processing Unit) technology provided by NVIDIA GRID processors. GPU virtualization provides robust rendering over Ethernet
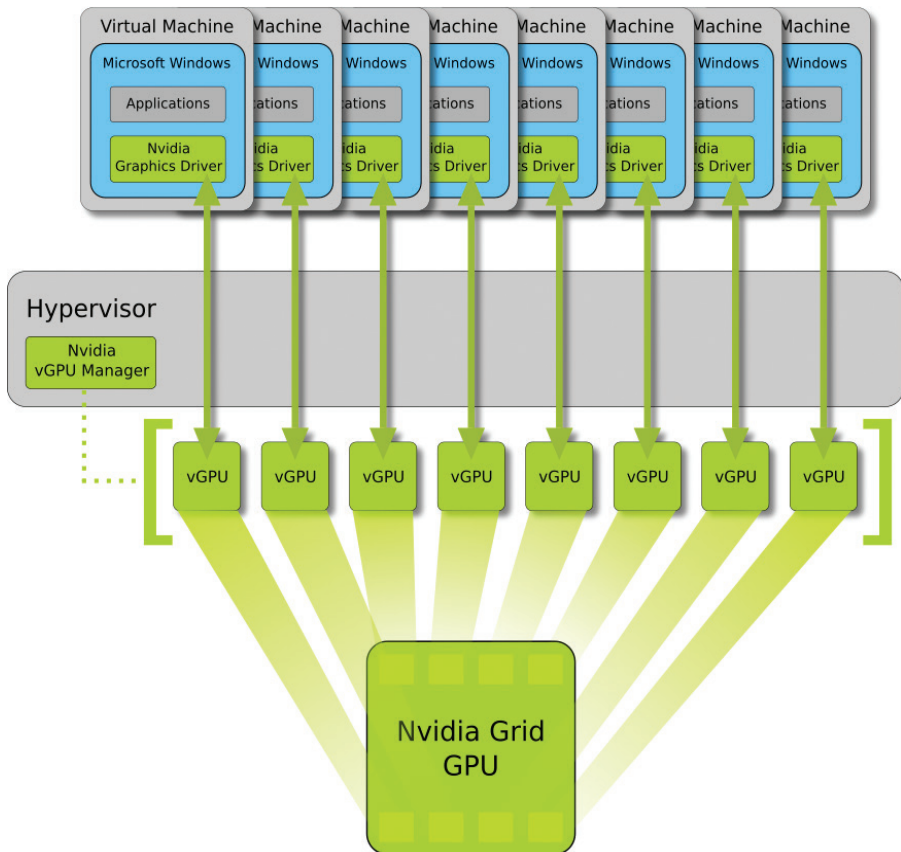

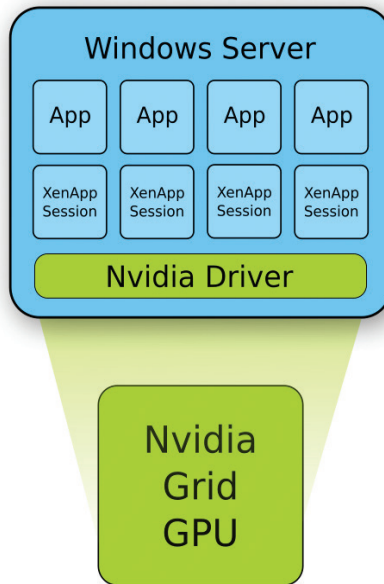**Figure 4.** Virtual desktop infrastructure (source: ICARUS).

Figure 5. Software as a Service architecture (source: ICARUS).

functionality and supports parallel computation with the CUDA framework. These functionalities are very promising technologies for mobile robotics applications where 3D maps derived from many sensors (3D lasers, photogrammetric cameras) have to be integrated. A very important aspect is that due to bandwidth limitations, the data transfer from robots to end users must be limited. To reduce the data flow, the rendering of such data is performed on the server, and only images (from the 3D rendering) are streamed over the network. This approach efficiently reduces the needed bandwidth for interaction with maps.

4.1. Virtual desktop infrastructure

The VDI technology allows operating system (in this case Win7) virtualization with the virtual hardware assignment per user. In this scheme, it is possible to allocate proper hardware resources needed for certain users. VDI technology allows vGPU sharing with many users simultaneously. Thus, users can be equipped with a full GPU (GPU Pass-Through mode), ½ GPU (2 users per GPU), ¼ GPU (4 users per GPU) or 1/8 GPU (8 users per GPU). It can be noticed that most demanding VDIs (simulation of UGV and simulation of USV) have been assigned a pass-through mode of GPU, allowing for the highest simulation performance. Less-demanding VDIs have been assigned only 1/8 GPU. It is possible to dynamically adjust the GPU placement policy to different needs and applications. The advantage of VDI is that

once assigned, virtual hardware belongs only to the user and other users do not affect it. This functionality guarantees a flexible and stable simulation environment for demanding training tools.

4.2. Integration of training and support with ICARUS system

Figure 6 shows data flow from/to the Remote Command and Control System (RC2, see chapter 8 of this book and [10]) to/from the Training and Support System (in training mode). The goal was to provide an interface to the RC2 via control interfaces. The final communication emulation module translates the ICARUS interface into the internal Training and Support communication scheme. There are two modes: training mode and support mode. During the training mode simulation, training and support tools are integrated with the ICARUS system to provide the training capabilities based on the real or the virtual components of the ICARUS system. **Figure 7** shows the support mode where the support tools are integrated with the ICARUS system. In this mode, the operator has access to additional information provided by the Command and Control Component via the Human Machine Interface.
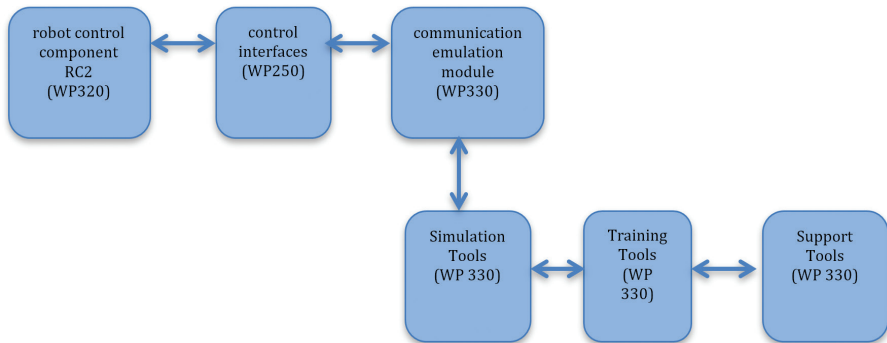

Figure 6. Data flow from/to RC2 to/from the training and support system (in training mode) (source: ICARUS).
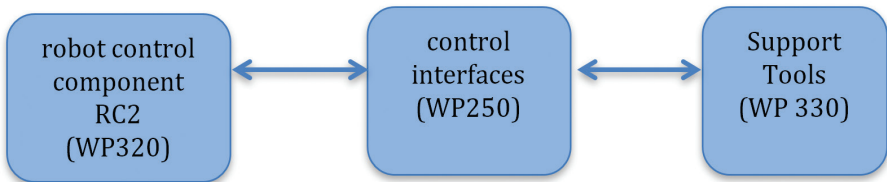

Figure 7. Data flow from/to RC2 to/from the training and support system (in support mode) (source: ICARUS).

5. Communication emulation module

The communication emulation module (**Figure 8**) is responsible for the simulation of the properties of an existing, planned and/or nonideal network with a certain propagation model. It is possible to simulate several nodes with a realistic radio propagation model among themselves that could be chosen depending on the situation of the network. The communication module mainly emulates the common attributes in a typical network, such as the packet losses and the delay on the established connection between the Unmanned Vehicle and the Operator (RC2).

To determinate the network behaviour, this module receives telemetry data from Unmanned Vehicles, containing information regarding their position and expected traffic load, supposing that this module controls the rest of the nodes in the network. That is why this module is composed mainly of two components:

- Network Emulator: The aim is to emulate the network conditions such as topology, congestion, load balance, node failure and so on that could vary depending on, for example, the position of the UV, new obstacles or environmental conditions.

- RF Emulator: The aim is to emulate the radio propagation models to predict the received signal in the network nodes. The wireless technologies which will be used between UV and RC2 are 802.11 and DMR.

The telemetry data (position, speed, yaw, pitch, roll, etc.) from the Unmanned Vehicle Simulator are received at the RF emulator in real time, that is without delay. From this information,
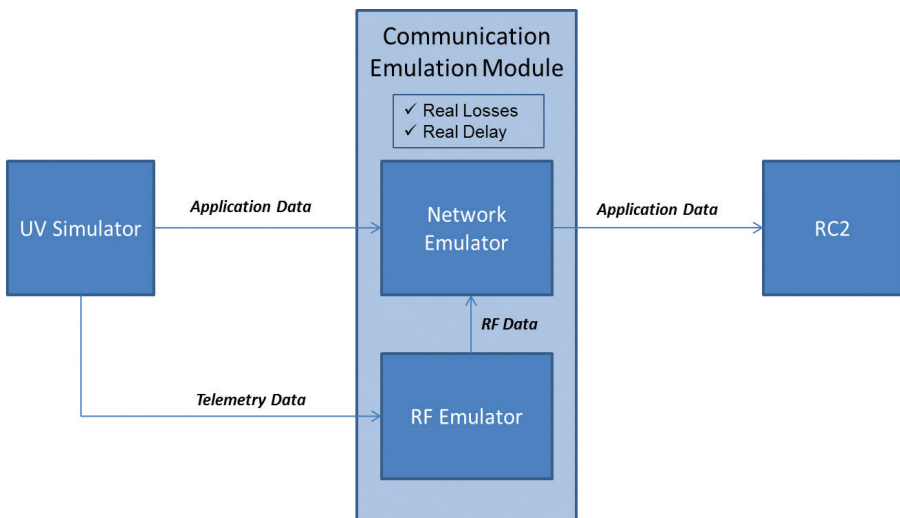

Figure 8. Communication Emulation Module (source: ICARUS).

the RF Emulator applies the appropriate propagation model and calculates the link budgets among possible network links. We suppose that the only node with movement would be the Unmanned Vehicle, although the rest of the nodes could also cause changes in the link budgets.

6. Network emulator

The Network Emulator module is based on UML (User-Mode-Linux), for instance, nodes in the network. UML allows starting a Linux Machine as a user process running within the host machine. From the point of view of the host machine, UML is a normal user process. From the point of view of a process that runs within UML, UML is a kernel, offering virtual memory and accessing to devices, what we call virtual machine. The UML architecture is shown in **Figure 9**.

The UML Kernel does not communicate directly with the hardware; it does it through Linux Kernel of the host machine. The processes that run within UML work the same way that within the Linux real machine, so that UML offers its own addressing space of kernel and process, its system of management of memory and planning.

The file system that UML uses to start each virtual machine is stored in a single file. This file is where the kernel of Linux and the configuration of a virtual machine are found. When we want to start a virtual machine, UML starts it basing on the kernel installed in that system file.

Thanks to the UML, we can execute a group of virtual machines within a real machine, the host machine. The virtual machines are connected to virtual collision domains. A virtual machine can work like a terminal machine, a router or a switch. Network Emulator module is developed using UML technology, thus a group of commands allows configuring and connecting virtual machines and the file system.

The devices may incorporate a varying amount of standard network attributes like: the round-trip time across the network (latency), the amount of available bandwidth, a given degree
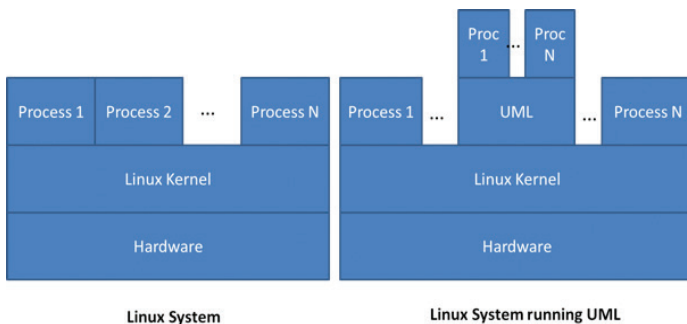

Figure 9. UML architecture (source: ICARUS).

of packet loss, duplication of packets, reordering of packets, corruption and modification of packets and/or the severity of network jitter. It can also mimic typical Layer 1 physical errors, such as Bit Error Rate, Loss of Signal, Output Bit Rotation and others.

7. Validation of the 3D-modelling capabilities of the support system

7.1. Validation in a marine incident scenario

The main feature of the support system is the ability to quickly merge raw 3D scans into complete 3D models of the environment. The 6DSLAM algorithm used for this task was tested on several different occasions in various environments. A major test of mapping capabilities was performed during the Icarus Sea Demo in Lisbon, Portugal. During the trial, a 3D model of the area where the trial took place was made by a robot shown in **Figure 10**—a husky robot with rotating SICK 500 LMS and a LadyBug 3 spherical camera. The final model is shown in **Figure 11** . The created model consists of over 90 million points.

7.2. Validation during euRathlon 2015 multirobot multidomain competition

The support system was also used during the euRathlon 2015 competition [11]. The 3D mapping system was mounted on one of the Icarus team's robots: Teodor. During robot operation the system was gathering 3D data about the mission area. It was also the key element of semiautonomous operation of the robot, as 3D data was used for planning the motion of the robot.

Figure 10. Dedicated support system—mobile mapping platform (source: ICARUS).
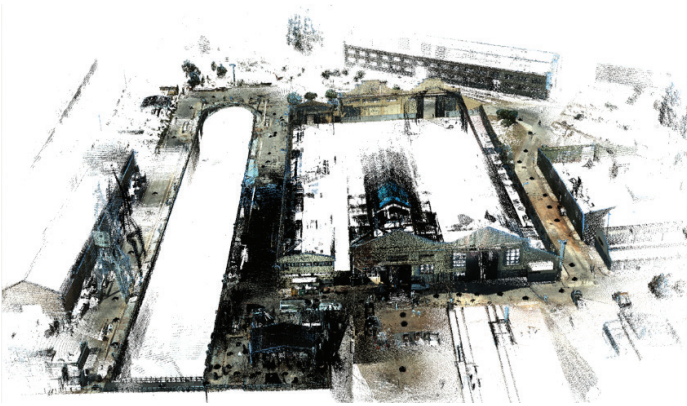
Figure 11. Initial environment model from Sea Trials (source: ICARUS).

The three main areas in which the support system was used during the operation was creating 3D maps of the environment, finding objects of interest and allowing semiautonomous operation of the robots.

The 3D mapping capabilities of the support system were used to create an outdoor map of the area. The map was coloured based on data from the ladybug. For the final scenario, the Grand Challenge, the created land map was merged with a 3D map obtained from an unmanned aerial vehicle to create a multilayer complex map of the area (**Figure 12**). The result was highly praised by the judges.

Using a 360° camera allowed to find a number of objects of interest undetected by the operator and the automatic algorithms connected to classical robot camera (**Figure 13**). Apart from

Figure 12. Multilayer map of euRathlon Grand Challenge area; top: land map layer, bottom: aerial map layer (source: ICARUS).
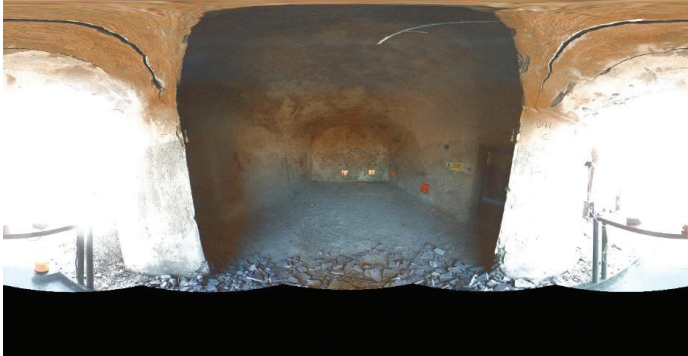
Figure 13. Ladybug spherical image with a number of OPIs (source: ICARUS).

a wider field of view, tools available in the support system allowed to enhance the gathered images in post-processing. After this process, some objects of interest that were previously not visible in raw data (and as such were not detected by either the operator or the automatic algorithm) became visible (**Figure 14**).

Figure 14. Enhanced LadyBug image. The marker inside the house is visible (source: ICARUS).

8. Conclusion

In this chapter, the ICARUS Training and Support system is discussed, introducing the High-Performance Computing (HPC) in the Cloud concept for improving the command and control system. The integration and visualization of maps derived from the different unmanned vehicles is the main functionality of this system. These maps are made available in the Cloud by the presented system, such that all the information concerning the disaster can be distributed over Ethernet, while respecting bandwidth limitations. A second important functionality is the serious games in the Cloud which are used for training end users. The proposed HPC solution is capable of streaming serious games over Ethernet, which means that using this system, international search and rescue teams can train simultaneously from offices in different countries.

Acknowledgements

Author details

Janusz Będkowski[1]*, Karol Majek[1], Michal Pełka[1], Andrzej Masłowski[1], Antonio Coelho[2], Ricardo Goncalves[2], Ricardo Baptista[2] and Jose Manuel Sanchez[3]

*Address all correspondence to: januszbedkowski@gmail.com

1 Institute of Mathematical Machines, Warsaw, Poland

2 DEI, Faculdade de Engenharia, Universidade do Porto/INESC TEC, Porto, Portugal

3 IntegraSys S.A., Las Rozas, Spain

References

[1] Kruijff GM, Colas F, Svoboda T, van Diggelen J, Balmer P, Pirri F, Worst R. Designing intelligent robots for human-robot teaming in urban search and rescue. In: AAAI Spring Symposium Series on Designing Intelligent Robots; AAAI Publications; 2012

[2] Dias PS, Gomes RMF, Pinto J. Mission planning and specification in the Neptus framework. In: IEEE International Conference on Robotics and Automation (ICRA); 15-19 May; IEEE; 2006. pp. 3220-3225. DOI: 10.1109/ROBOT.2006.1642192

[3] Hamp Q, Gorgis O, Labenda P, Neumann M, Predki T, Heckes L, Kleiner A, Reindl L. Study of efficiency of USAR operations with assistive technologies. Advanced Robotics. 2013;**27**(5):337-350

[4] Murphy RR. Trial by fire [rescue robots]. Robotics Automation Magazine. 2004;**11**(3):50-61

[5]   Murphy RR, Tadokoro S, Nardi D, Jacoff A, Fiorini P, Choset H, Erkmen AM. Search and rescue robotics. In: Siciliano B, Khatib O, editors. Handbook of Robotics. Springer-Verlag; 2008. pp. 1151-1173. DOI: 10.1007/978-3-540-30301-5_51

[6]   Liu Y, Nejat G. Robotic urban search and rescue: A survey from the control perspective. Journal of Intelligent & Robotic Systems. 2013;**72**(2):147-165

[7]   Doroftei D, De Cubber G, Chintanami K. Towards collaborative human and robotic rescue workers. In: Human Friendly Robotics; 2012

[8]   Marques MM, Martins A, Matos A, Cruz N, Almeida JM, Alves JC, Lobo V, Silva E. REX14—Robotic Exercises 2014—multi-robot field trials. In: MTS/IEEE OCEANS 2015; Washington, DC, USA. 2015. pp. 1-6. DOI: 10.23919/OCEANS.2015.7404497

[9]   Balta H, Bedkowski J, Govindaraj S, Majek K, Musialik P, Serrano D, Alexis K, Siegwart R, De Cubber G. Integrated data management for a fleet of search-and-rescue robots. Journal of Field Robotics. 2016;**34**(3). DOI: 10.1002/rob.21651

[10] Govindaraj S, Chintamani K, Gancet J, Letier P, Van Lierde B, Nevatia Y, De Cubber G, Serrano D, Bedkowski J, Armbrust C, Sanchez J, Coelho A, Palomares ME, Orbe I. The ICARUS Project—Command, Control and Intelligence (C2I). In: Safety, Security and Rescue Robots; October 2013; Sweden: IEEE; 2013

[11] Marques MM, Parreira R, Lobo V, Martins A, Matos A, Cruz N, Almeida JM, Alves JC, Silva E, Będkowski J, Majek K, Pełka M, Musialik P, Ferreira H, Dias A, Ferreira B, Amaral G, Figueiredo A, Almeida R, Silva F, Serrano D, Moreno G, De Cubber G, Balta H, Beglerović H, Govindaraj S, Sanchez JM, Tosa M. Use of multi-domain robots in search and rescue operations—contributions of the ICARUS team to the euRathlon 2015 challenge. In: IEEE OCEANS; April; Shanghai, China: IEEE; 2016

[12] Susi T, Johannesson M, Backlund P. Serious Games: An Overview. 2007

[13] Available from: http://www.businessdictionary.com/definition/training.html

[14] Sugar S, Whitcomb J. Simple and Effective Techniques to Engage and Motivate Learners. American Society for Training and Development (ASTD); 2006

[15] Sawyer B, Smith P. Serious game taxonomy. In: Paper presented at the Serious Game Summit 2008; San Francisco: USA; 2008

[16] Available from: https://www.americasarmy.com

[17] Carpin S, Lewis M, Wang J, Balakirky S, Scrapper C. USARSim: A robot simulator for research and education. In: IEEE International Conference on Robotics and Automation; 2007. pp. 1400-1405. DOI: 10.1109/ROBOT.2007.363180

[18] Available from: http://unity3d.com

[19] Available from: http://moast.sourceforge.net

[20] Olivier, M. / Cyberbotics Ltd - Webots TM: Professional Mobile Robot Simulation, Inernational Journal of Advanced Robotic Systems, 2004;**1**(1): ISSN 1729-8806 pp. 40-43

[21] Available from: http://www.ede.org

[22] Available from: http://wiki.ros.org/gazebo

Operational Validation of Search and Rescue Robots

Geert De Cubber, Daniela Doroftei, Haris Balta,

Anibal Matos, Eduardo Silva, Daniel Serrano,

Shashank Govindaraj, Rui Roda, Victor Lobo,

Mário Marques and Rene Wagemans

Additional information is available at the end of the chapter

Abstract

This chapter describes how the different ICARUS unmanned search and rescue tools have been evaluated and validated using operational benchmarking techniques. Two large-scale simulated disaster scenarios were organized: a simulated shipwreck and an earthquake response scenario. Next to these simulated response scenarios, where ICARUS tools were deployed in tight interaction with real end users, ICARUS tools also participated to a real relief, embedded in a team of end users for a flood response mission. These validation trials allow us to conclude that the ICARUS tools fulfil the user requirements and goals set up at the beginning of the project.

Keywords: rescue robotics, operational validation

1. Introduction

As shown in the previous chapters of this book, the ICARUS project developed multiple unmanned systems and tools for supporting search and rescue (SAR) teams. These technological tools were developed after a careful consideration of the end-user needs [1], as discussed in Chapter 2 of this book. Of course, during and at the end of the design lifecycle, the performance of the different tools with respect to the user requirements and target performance levels needs to be evaluated. This process of system validation requires a careful compromise between two points of view:

INTECH
open science | open minds

1. End users want validation tests to occur in realistic operational conditions, mimicking as closely as possible a real deployment.

2. Scientists want validation tests to have statistical relevance, so they want repeated tests, performed under controlled environments. However, it is very hard to quantify the system performance in a rigorous scientific manner due to the fact that many variables are out of control in an outdoor environment, e.g. the weather conditions (wind, rain, sea state, illuminance, etc.). Moreover, a scientific evaluation requires that multiple trials must be held to validate the statistical significance of the quantitative results, which is not evident when confronted with the evaluation of complex heterogeneous robotic teams in operational conditions, requiring significant logistics for setting up each trial run.

In the past, multiple proposals have been made in order to combine these different points of view [2]. As a result, validation methodologies can be generally categorized into two approaches:

1. The first approach is based on the development of highly standardized test methodologies [3]. A good example is that developed and proposed by the National Institute of Standards and Technology (NIST). The big advantage of these methodologies is that they allow to accurately quantify the performance of the robots in a number of test setups. The disadvantage of these methods is that, due to their standardized nature, these approaches are often quite dissociated from practical operational conditions.

2. The second approach for validation is robot competitions like DARPA [4], euRathlon [5] and ELROB [6]. Here, multiple robotic systems are pitted against each other in more or less realistic operating conditions. The advantage of this validation approach is that the performance can be evaluated in real-life like circumstances and environments. The disadvantage of these kinds of benchmarking methodologies is that, due to their non-standardized nature, they often only allow a qualitative appreciation of the robot performance and do not allow making a detailed quantitative measurement. Another important disadvantage is that coincidence (e.g. dependence on singular element failures that may not be exemplar for the overall system operation, changing weather and lighting conditions between trial runs, etc.) plays an important role in these competitions, which significantly compromises the statistical significance of the benchmarking result.

It is clear that both of these approaches are highly valuable and necessary. However, none of them gives an ultimate solution for the performance evaluation problem. Here, we present the operational test and validation approach for the evaluation of the performance of a range of marine, aerial and ground search and rescue robots. This methodology has been proposed and followed within the ICARUS project. The proposed approach aims to find a compromise between the traditional rigorous standardized approaches and the more open-ended robot competitions. Following this methodology, operational scenarios are defined that include a performance assessment of individual robotic tools. Furthermore, these operational scenarios also assess the performance of heterogeneous teams of robotic tools, cooperating not only among robots, but also with manned teams in realistic search and rescue activities. In this way, it is possible to perform a more complete validation of the use of robotic tools in challenging real-world scenarios.

The ICARUS project considers two main demonstrations to validate the tools developed during the whole duration of the project [7]:

1. A marine demonstration, simulating a shipwreck in coastal waters.

2. A land demonstration, simulating an earthquake in an urban environment.

This chapter reports on the results of the operational validation performed during both demonstrations. However, the proof of the pudding is in the eating. No simulated disaster management exercise can mimic the chaos and difficult environmental conditions encountered during a real response operation. Therefore, we also included a section reporting on the use of one of the unmanned aerial systems deployed during a real flood-relief operation, within the framework of the ICARUS project.

2. Maritime demonstration: simulated shipwreck response

2.1. General storyboard

Organized in coordination with the yearly REX exercises [8], the scenario for the maritime demonstration is based on a shipwreck of a ferryboat. For such a scenario, the roles of the robotic tools can be separated according to the nature of the platforms. Aerial vehicles are used in search operations, sweeping the area and providing information about the exact location of the accident, localizing victims on the water and tracking them. Another role of the aerial segment is to carry mobile communication equipment allowing for the establishment of an extended range mobile network to support communications between all robotic assets and also with manned platforms. On the other hand, the main role of maritime platforms is on the rescue operations. Taking advantage of the data collected by the aerial segment, maritime platforms get close to located victims and assist them. Such assistance consists in providing floatation and shelter from environmental conditions, extending their time of life and allowing rescue in safe conditions.

2.2. Location and organisational issues

The sea demonstration took place on the Tagus river estuary in the area of the Lisboa Naval Base, located in Alfeite, Almada, Portugal, as shown in **Figure 1**. The selection of this place took into account several issues related to the demonstration: segregated area for the operation of unmanned systems, realistic scenario for a search and rescue operation, easy access to the operational area and for mounting command and control stations as well as communications equipment.

Figure 1 shows the location of the ICARUS sea demonstration area, off the south bank of Tagus river estuary. This figure also exhibits the navigation lanes of ferryboats that continuously cross the estuary, a characteristic that was taken into account in the selection of a realistic location for the demonstration. Furthermore, the selection of a location right in front of the Lisboa Naval Base simplified the logistics associated to the operation due to the existence of local facilities from the Portuguese Navy and from the Arsenal do Alfeite Shipyard.

Figure 1. Location of the sea demonstration area: Lisboa Naval Base in Portugal, showcasing the area reserved for the ICARUS sea demonstration (source: ICARUS).

For the operations on the water, the region shown in **Figure 1** was closed to maritime traffic, except for the vessels required for the demonstration and to patrol the area to make sure that unauthorized vessels do not enter it, so that all activities related to the demonstration could be carried out without major concerns with other activities that could cause any kind of interference. This area exceeds 2 km² and the maximum distance from shore is about 2.2 km.

2.3. Operational validation for maritime search and rescue

Time: T0

Events: An explosion of unknown origin occurs in a ferryboat crossing the Tagus river estuary near Lisboa. Victims fall in the water. The ferryboat, shown in **Figure 2**, starts sinking.

Figure 2. Ferryboat used for simulating the shipwreck accident (source: ICARUS).

Time: T0 + 0h05min

Events: The Marine Rescue Coordination Center (MRCC) in Lisboa receives an alert describing the accident and its approximate location. After an initial assessment, the MRCC dispatches to the area search and rescue teams (including robotic assets).

Time: T0 + 0h40min

Events: Search and rescue teams start arriving at the location of the accident (riverbank). A local coordination centre, shown in **Figure 3**, is set up where the ICARUS command and control system [9] is deployed and communications with MRCC are established. Search and rescue assets, including manned rigid-hulled inflatable boats and unmanned aerial and surface vehicles, are prepared for launching.

Evaluation: For a search and rescue mission where every minute counts, the set-up time for the developed technological tools is still an important factor where progress can be made. Confronted with the complicated frequency spectrum in the Lisbon harbour environment, it was mainly the configuration of the communication tools which increased the overall set-up time.

Figure 3. Set-up of ICARUS command and control tools at the local coordination centre. (source: ICARUS).

Time: T0 + 1h00min

Events: An area around the approximate location of the accident is defined and a fixed wing long endurance unmanned aircraft is deployed to make a survey of that area, as shown in **Figure 4**.

Evaluation: Take-off and landing operations of the fixed wing long endurance unmanned aircraft were performed manually by trained personnel. As the fixed wing long endurance unmanned aircraft is hand-launched, take-off is still quite easy, even in a cluttered

environment. However, landing requires the careful choice of a suited landing position, which is tough in a heavily built-up harbour environment. Luckily, the fixed wing long endurance unmanned aircraft can stay airborne for multiple hours (even days), so the issue does not occur often. In flight, the fixed wing long endurance unmanned aircraft semi-autonomously executed a GPS-defined trajectory and a search pattern mission over the area defined for the sea demonstration. A person external to the operating team provided the profile of this mission. Several flight patterns were tested and successfully performed by the fixed-wing long-endurance unmanned aircraft.

Figure 4. Initial area surveillance with the fixed wing long endurance unmanned aircraft (source: ICARUS).

Time: T0 + 1h10min

Events: Images collected by the fixed wing long endurance unmanned aircraft start arriving at the local coordination centre and at the MRCC, providing information about the location of the ferryboat, victims on the water and debris scattered over the area, as shown on **Figure 5**.

Evaluation: Live streams from the thermal and video camera on-board the fixed wing long endurance unmanned aircraft were received at the command and control interface. Victim positions were also transferred to the command station.

Time: T0 + 1h20min

Events: Based on the information collected by the fixed wing long endurance unmanned aircraft, the rescue operation is planned at the MRCC: areas of intervention are assigned to manned rigid-hulled inflatable boats and carrier unmanned surface vehicles. The MRCC

sends operation plans to local coordination centre; these plans are received by the command and control station of robotic assets and rescue operations start.

Evaluation: In order to test the detection capabilities of the different assets, the victims in the water were asynchronously spread over multiple clusters: there were clusters with one person, two persons and four persons. The relatively high-altitude fixed wing long endurance unmanned aircraft proved to be able to spot the clusters of victims in the water, but was

Figure 5. Information being displayed on the command and control interface (source: ICARUS).

unable to count the number of victims per cluster, due to limitations in sensor resolution and the high flight altitude.

Time: T0 + 1h30min

Events: Carrier unmanned surface vehicles head to clusters of victims located further away; at the same time, the rotary wing unmanned aircraft is launched to track victims that will be rescued by unmanned capsules on-board two carrier unmanned surface vehicles: the ROAZ II and the U-RANGER, as shown on **Figure 6**. Manned rigid-hulled inflatable boats also depart to rescue people closer to the margin, while another unmanned capsule is deployed and remotely piloted towards victims that are close to the margin.

Evaluation: Similar to the endurance aircraft, the take-off and landing of the rotary wing unmanned aircraft were performed manually and mapping and victim search were executed semi-autonomously. For legal and safety reasons, the U-RANGER always operated with a person on-board. Besides the command and control interface that remotely operated the unmanned surface vehicle, a second backup control station was used to create another safety loop. Contrary to the fixed-wing long-endurance unmanned aircraft, the rotary-wing unmanned aircraft proved to be capable of not only detecting the clusters of victims but also of counting the number of victims per cluster, which provided important information for the allocation of resources. Space management between manned and unmanned assets was an important factor in this phase of validation, as multiple simultaneous rescue operations started in the same area. This incurred that the U-RANGER could not operate at full speed due to safety reasons, but due to the intelligent obstacle avoidance capabilities of the unmanned systems, no problems occurred.

Figure 6. Rotary wing unmanned aircraft providing victim's location to ROAZ II (source: ICARUS).

Time: T0 + 1h40min

Events: One carrier unmanned surface vehicle arrives near a first cluster of victims and deploys one unmanned capsule, as shown on **Figure 8**. This unmanned capsule is remotely operated to move towards the victims using location information provided by the rotary-wing unmanned aircraft, as shown on **Figure 7**. When the unmanned capsule arrives close to the victims, as shown on **Figure 9**, it inflates a life raft and victims start climbing on-board the life raft.

Figure 7. Rotary wing unmanned aircraft providing victim location to an unmanned capsule (source: ICARUS).

Evaluation: Once started, the rescue operations moved very quickly. During the whole operation, live video feeds from thermal and visible cameras were received at the base station console. Radar and laser data used to detect obstacles on the water were also transmitted to shore.

Time: T0 + 1h45min

Events: The same carrier unmanned surface vehicle moves now towards another cluster of victims. The rotary wing unmanned aircraft tracks the location of these victims that are drifting away due to the water current. Another unmanned capsule is launched and automatically moves towards the victims using information provided by the rotary wing unmanned aircraft. Again, it inflates the life raft when close to the victims. Meanwhile, the other unmanned surface vehicle also deploys an unmanned capsule; this unmanned capsule moves autonomously towards a location where victims were spotted; the life raft is inflated by direct action of the victims. By that time, the unmanned capsule launched from the riverbank already reached the victims and its life raft is inflated.

Figure 8. Unmanned capsule being launched from ROAZ II (source: ICARUS).

Figure 9. Victim being rescued by an unmanned capsule (source: ICARUS).

Evaluation: Simultaneous rescue operations normally provide a cognitive overload for the commander in charge of the operation, but thanks to the ecologic display functionalities of the ICARUS command and control system, the commander could keep an overview of the different operations and coordinate the instructions towards the different team members.

Time: T0 + 1h50min

Events: The carrier unmanned surface vehicles head now to the area behind the sinking ferryboat, as shown on **Figure 10**; thermal and visible images are sent back to the local coordination

Figure 10. U-RANGER and ROAZ II providing situational awareness to rescue workers behind the ferryboat (source: ICARUS).

centre for situation assessment. The aerial vehicles provide wireless links to the unmanned surface vehicles while they are operating out of sight.

Evaluation: The ferryboat blocked line of sight connectivity to the unmanned surface vehicles. However, the fixed wing long endurance unmanned aircraft acted as a relay station, providing connectivity to both vehicles, allowing for streaming real-time video to the base station.

Time: T0 + 2h00min

Events: Information received from the unmanned surface vehicles is then used to plan a rescue operation for people still on the ferryboat. By that time, the robotic assets are recovered and their activities end.

2.4. Conclusions of the marine demonstration

Besides the impact of the sea scenario demonstration in the media and the opportunity it provided for establishing contacts with stakeholders and other relevant players, the experiments conducted during the trials were used to obtain qualitative and quantitative information about the technical developments of the project and the way they met the established goals.

Such assessment was organized along the following lines:

1. Validation of capabilities. During the experiments, capabilities (sets of aggregated requirements) planned for each system or set of systems were assessed. A total of 71 of 84 capabilities were validated.

2. Performance analysis. Performance analysis consisted in assessing quantitative metrics against three performance targets (minimum acceptance, goal and breakthrough levels). For the total of 66 metrics considered, 36 (55%) reached the breakthrough level, 18 (27%) reached the goal level, 11 (17%) reached the minimum acceptance level and just 1 (2%) was below this minimum level.

Two officers from the Portuguese Navy were asked to evaluate the individual platform experiments, either when they were conducted independently or while they were carried out as part of more complex tests. In each case, they were asked to score the experiment and provide some comments. The feedback provided were extremely positive and in line with the other evaluation methods.

3. Land demonstration: simulated earthquake response

3.1. General storyboard and setting

In order to validate the performance of the ICARUS tools in an urban search and rescue context, the ICARUS land demonstration defines an earthquake-response scenario where the different ICARUS aerial and ground assets are used to support the relief teams. The ICARUS land demonstration was integrated into a training exercise of the Belgian First Aid and

Support Team (B-FAST) as a preparation for their INSARAG IEC re-classification tests. As such, the complete integration of unmanned tools in the standard operating procedures of real search and rescue workers could be tested.

The ICARUS land validation took place in the military base Camp Roi Albert in Marche-en-Famenne, Belgium, a woody and hilly area halfway between Brussels and Luxemburg. The base is the regular training ground for the B-FAST team and provides for this purpose a rubble field with a pancake house for performing victim search and rescue operations and a built-up area which can serve as a mock-up urban setting for testing urban search and rescue protocols.

Figure 11 shows the different areas within the simulated crisis area which were used throughout the operations:

- The Base of Operations (BoO) set up by the B-FAST team.

- The location of the city of Focagne, which is the urban area assigned to the B-FAST team.

- The Forward BoO which is set up by B-FAST close to the Focagne city.

- The road from the BoO to the Forward BoO which is partially blocked by debris.

- Apartment buildings which have collapsed and where victims could be found in the voids between the rubble.

- A semi-demolished school building where trapped school children are present.

- A warehouse on fire with chemical products inside.

In the rest of this section, we will explain how the different ICARUS tools assist the search and rescue workers in dealing with each of the presented difficulties.

Figure 11. Situational overview of the crisis area (source: ICARUS).

3.2. Integration of the ICARUS system into the OSOCC

Events: As shown on **Figure 12**, the B-FAST Urban Search and Rescue (USAR) team sets up the On-Site Operations Coordination Centre (OSOCC)-level C4I equipment which includes a large workstation with displays. The USAR team's communication specialist sets up local communication equipment and tests to see if web access and local GSM networks are available. The ICARUS C4I systems then connect to the Global Disaster Alert and Communication System (GDACS) and pulls in the latest data about the disaster.

Figure 12. OSOCC set up by the B-FAST team, serving as base command station for the operations (source: ICARUS).

Evaluation: The ICARUS command and control system was successfully connected to the OSOCC and imported GIS data, fact sheet data and GDACS data about the disaster. The end users did evaluate the total setup time of the whole system (1 hour) as still too slow. The main issue is communication, which is of course not an easy parameter to quantify and debug, as the communication ecosystem and frequency spectrum usage will be different in every crisis.

3.3. Support to mission planning

Events: As very little information is available, the planning officials of the team decide to use unmanned tools to obtain quickly a better common operational picture of the situation. The B-FAST team leader orders the fixed wing endurance aircraft to scan the city of Focagne, as shown on **Figure 13**. The human operator at the BoO selects a geo-referenced scan on the human-machine interface, after which the aircraft executes the task semi-autonomously.

Figure 13. Launching the endurance aircraft during the demonstration (source: ICARUS).

The goal of this first scan is to do obtain a good overview of the level of destruction in Focagne.

Evaluation: The UAS autonomously acquired data (visual + IR imaging) over the area of interest and transmitted this data in real time to the base station. The data from the UAS were used to reconstruct a map, which the mission planner overlapped with the pre-existing GIS data. Based on the obtained information, the mission planner used the command and control system for sectorization. More detailed scans per sector were then performed.

End users were very impressed with the speed of obtaining a high-quality situational overview of the crisis area using the data gathered by the aircraft via the ICARUS command and control system and many positive remarks were voiced.

3.4. Deployment of the USAR team

Events: As shown on **Figure 14**, the USAR teams move towards and deploy into a sector assigned by the mission planner via the command and control system. The main purpose of this scenario is to test the (rapid) deployment capabilities and the integration of the communication and command and control system. Another purpose of this scenario is to test the network and command and control system management capabilities when confronted with dynamic team and resource allocations and to test the capability of the aircraft to detect roadblocks.

Figure 14. B-FAST team advancing with all ICARUS tools: large UGV (driving in front), small UGV (packed on first vehicle), rotorcraft (airborne in the middle of the picture) and fixed wing aircraft (airborne, but not visible in this picture) (source: ICARUS).

Evaluation: The B-FAST team moved from the base of operations towards the forward base of operations, together with all the ICARUS tools. Organising this scenario required convincing the end users of the added value the ICARUS tools could bring in this phase of the operation. Indeed, in the beginning, the end users were afraid that the ICARUS tools would needlessly delay the USAR deployment operation, whereas speed is a key issue of course in any search

and rescue operation. For this deployment operation, mostly the large UGV posed issues as it is of course not as fast as a standard truck. The ICARUS team therefore worked hard on finding the right balance between teleoperation and autonomous guidance for driving the large UGV as fast as possible up the hill without slowing down the B-FAST convoy. This succeeded very well, as the convoy could advance at a very normal speed during the public demonstration day, to the satisfaction of the B-FAST users. The B-FAST end users also highly appreciated the continuous live input from the outdoor rotorcraft, warning them about road blocks, which could save them valuable time in a real operation.

3.5. Victim search and rescue in demolished apartment buildings

Events: The USAR team rescues victims trapped in a semi-demolished apartment building, helped by the ICARUS UGV and UAV systems. The main objective of this scenario is to test the assessment, search and rescue capabilities of the outdoor rotorcraft and the large UGV and their collaborative operation mode.

The fixed wing aircraft is sent to sector to perform long-range human detection using its infrared detector. It scans the area where the apartment buildings have collapsed. The UAS returned a map indicating the locations of potential victims. A mission to investigate the potential victim locations is transferred to the rotorcraft which is sent out to provide a high-resolution 3D assessment of the site and to confirm the victim detections using its on-board human detector, as shown on **Figure 15**. The rotorcraft returns from its mission. A very high-resolution 3D map of the scan area is transferred, confirming the position of one undetected victim. The rotorcraft is sent out again to the victim location to assess the medical state of the victim. Analysis of the 3D map, imagery data and the victim location and medical state returned by the rotorcraft helps the planning team in setting up a plan to rescue the victims. Human rescue team members are sent out to rescue victims which can be evacuated without the help of the unmanned tools. The rotorcraft is (manually) equipped with a rescue kit and is requested to deliver this rescue kit to a victim which is trapped in the middle of the remains

Figure 15. Simultaneous aerial victim search operations at different altitudes (source: ICARUS).

Figure 16. Rescue kit delivery (source: ICARUS).

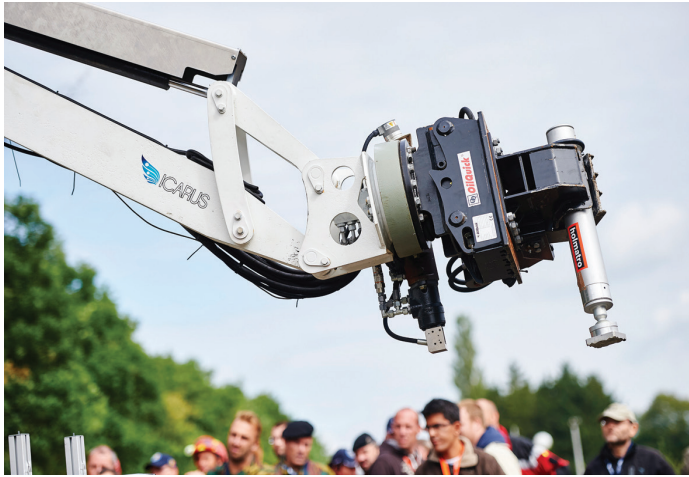**Figure 17.** Debris clearance with the large UGV (source: ICARUS).

Figure 18. Shoring operation with the large UGV (source: ICARUS).

of the demolished building, as shown on **Figure 16**. The large UGV has cleared a pathway to the victim, as shown on **Figure 17**. The human rescue team comes in and evacuates the victim. The canine rescue team has located another victim below the rubble. However, due to structural instability, the access to the victim is considered too risky for direct human intervention. The canine rescue team sends the data on the victim location to the RC2 using their mobile devices and request the intervention of the large UGV for shoring the access path to the victim. Upon receiving this task, the large UGV first heads (remotely controlled) to the local command station to pick up few hydraulic/pneumatic struts. The large UGV is in place to start the shoring operation, with all tools at hand. The large UGV uses its manipulator arm, remotely controlled using an exoskeleton, to place the struts, as shown on **Figure 18**. The struts are activated remotely by a human operator. The large UGV has stabilised the entrance towards the sixth victim. Human rescue team members enter and evacuate the victim.

Evaluation: End users and stakeholders were impressed by the seamless integration and collaboration as a team of ICARUS tools in the B-FAST search and rescue toolkit. The complementarity between canine and unmanned aerial search teams was applauded as extremely useful for future SAR operations.

3.6. Victim search and rescue in semi-demolished school building

Event: The team now focuses its attention on the school building where multiple children are reported missing. The building is too unstable for human rescue workers to enter safely. Luckily, however, the access to this building is clear. As such, it is decided that the indoor rotorcraft and the small unmanned ground vehicle will be sent in simultaneously for assessing
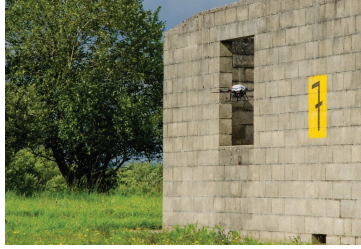
Figure 19. Indoor rotorcraft entering the building via broken window (source: ICARUS).

Figure 20. Unmanned ground vehicle navigating through the corridors of the building (source: ICARUS).

Figure 21. Indoor rotorcraft and small UGV exploring the school building (source: ICARUS).

Figure 22. Civil protection working acting as victim (obviously no real children could be used) (source: ICARUS).

the structural integrity of the building, giving rescue workers a view of what is happening inside and for finding survivors.

The indoor rotorcraft enters the building by flying through a broken window, as shown on **Figure 19**, while the small unmanned ground vehicle enters through the doors and corridors, as shown on **Figure 20**. Simultaneously, they explore and map the building, as shown on **Figure 21** both finding and localizing survivors with their infrared and visual sensors, as shown on **Figure 22**.

The structural scans by the unmanned systems help the planning team in deciding that the rescue team can be safely sent into the building to rescue the victims detected and localized by the unmanned systems.

Evaluation: End users were positively surprised about the navigation capabilities in small enclosures of both the rotorcraft and the small UGV. Specifically, the capability of the rotorcraft of flying through window openings only marginally larger than its proper size generated much appraisal. For the indoor operations which happened beyond line of sight, radio connectivity was obviously problematic, so the semi-autonomous exploration behaviours of both vehicles had to be used. While this led to a slower exploration strategy (notably for the ground vehicle) due to safety reasons, the whole planned operation could be completed within the time constraints.

3.7. Victim search and rescue in semi-demolished CBRN (Chemical, biological, radiological and nuclear) warehouse

Event: The mayor reports that there is a two-story warehouse with potentially dangerous products near the city centre. As previous attempts to enter the building without using the help of unmanned tools have failed and the structure is too unstable for the safe operation of humans near the walls, the help of all unmanned tools is requested for entering the building.

Figure 23. Unmanned tools navigating to CBRN warehouse (source: ICARUS).

Figure 24. Large UGV deploying small UGV on the first floor, assisted by the outdoor rotorcraft (source: ICARUS).

Figure 25. Remote operator using an exoskeleton to operate the manipulator arm of the small UGV (source: ICARUS).

Figure 26. Small UGV opening a door with the manipulator arm, remotely operated by the exoskeleton (source: ICARUS).

The large UGV is remotely controlled up to the building walls, as shown on **Figure 23**. With its manipulator arm, the large UGV deploys the small UGV which it was carrying onto a terrace on the first floor of the building, as shown on **Figure 24**. The small UGV moves to the door on the first floor. The exoskeleton is then used by a remote operator to control the manipulator arm on the small UGV to open the door, as shown on **Figure 25** and **Figure 26**. The small UGV is now inside the building and starts scanning the surroundings, searching for victims and dangerous products and inspecting the structural integrity of the building. The complete building is explored by the small UGV and two human survivors are detected and localized. Meanwhile, the large UGV is standing guard close to the building entrance to act as a wireless repeater, ensuring optimal communication to the small UGV inside and the outdoor rotorcraft is sent to explore the roof of the building to detect possible human survivors. The outdoor rotorcraft detects a human survivor on the rooftop. Visual inspection from the footage of the UAS also indicates that the emergency exits seem to be blocked, indicating that the survivors should be evacuated through the building. Based on the data provided by the rotorcraft and the small UGV, the rescue workers decide on the protective equipment to wear for entering the building and start the victim evacuation.

Evaluation: The public was impressed by the capability of the ICARUS command and control system to seamlessly manage the simultaneous operation of three unmanned tools (small UGV, large UGV and outdoor rotorcraft). Also the exoskeleton received a lot of attention and was praised by the spectators by its user-friendliness. Many stakeholders present at the demonstration knew (some also from their own experience) that opening a door with a robot manipulator without direct line of sight for the operator is a notoriously difficult operation and they could not believe how seemingly effortless this operation was achieved via the exoskeleton.

3.8. Conclusions of the land demonstration

Not less than five unmanned assets (large UGV, small UGV, fixed-wing aircraft, outdoor rotorcraft and indoor rotorcraft) collaborated during the different land demonstration

missions. All robotic assets were interfaced via the common ICARUS command and control interface, thereby showcasing the versatility and modularity of the ICARUS interoperability concept.

All victims who were hidden throughout the different missions were located in the end and received assistance, and a tight interrelation and even complete integration of ICARUS tools within the toolkit of the search and rescue workers was achieved.

As discussed before, a series of metrics and key performance indicators were defined by end users with detailed target performance levels to be attained by the different systems. For none of those metrics, a score was noted which formed a problem for global mission success. For 12% of the capabilities and metrics, the minimum acceptance level as imposed by the end users was reached, but the initially imposed target performance was not reached. For no less than 33% of the capabilities and metrics, the goal performance level as imposed by the end users was reached or even surpassed. Finally, for about half (47%) of the capabilities and metrics, consisting of the majority of the performance bins, a performance level which was indicated by the end users as "breakthrough-level" prior to the demonstrations was reached. This number is an extraordinary achievement, showcasing that the ICARUS project succeeded in surpassing by far the expectations of the end users.
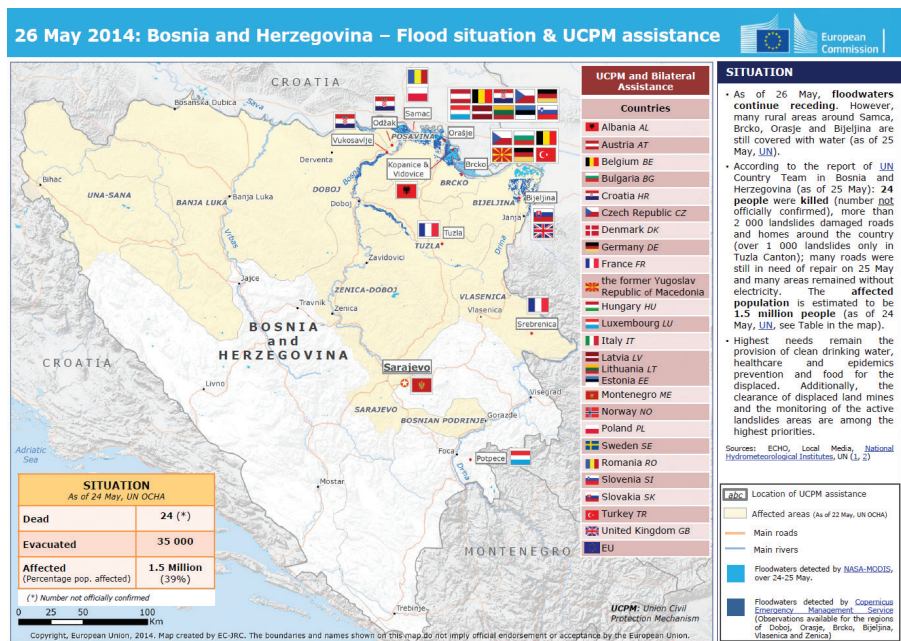

Figure 27. Bosnia flood map (source: European Union Joint Research Centre—source: European Union 2014—used with permission).

4. Real flood relief operation

4.1. Mission context

At the end of May and the beginning of June 2014, a catastrophic massive flooding occurred in Bosnia and Herzegovina, Croatia and Serbia due to abundant rainfall over the course of a few weeks. All countries suffered immense damage. In Bosnia and Herzegovina, where the whole northern region and part of the central region were heavily affected, it was estimated that 1.5 million people were affected (accounting to 39% of the population). Floods and landslides were responsible for at least 53 deaths in Bosnia and Herzegovina and Serbia [10].

In response to this catastrophe, the EU Civil Protection Mechanism was activated. Twenty-two EU member states offered assistance through the mechanism. **Figure 27** shows the flood situation and the deployment of international response teams, activated via the EU Civil Protection Mechanism. However, the relief efforts were hampered by the destroyed infrastructure, broken telecommunications, blackouts, etc. [10].

Making matters worse, Bosnia and Herzegovina was contaminated with landmines due to the war that took place there from 1992 to 1995, and as a result, the country has one of the most serious landmine problems in the world. The presence of many explosive remnants of war (ERWs) remaining from the Balkan War of the 1990s created a very dangerous situation for the relief workers and the local population. The floods, torrents, landslides and land-shifting had a destructive impact on the (previously mapped) suspected hazardous areas (SHAs) and minefields in Bosnia and Herzegovina. In Bosnia, 831.4 km² of SHAs were flooded, and 37.48 km² of SHAs in 33 locations were under direct impact of landslides and torrents. Due to the floods, the ERW started moving place and the SHAs had to be extended dramatically. By the 4th of July 2014, 1018 pieces of unexploded ordnance (UXO), 92 mines and 3 cluster bombs were already found, as well as 40,163 pieces of ammunition. Moreover, 80.2 km² of new areas that were previously not been suspected of having mines became potentially hazardous (mainly in the northern part of Bosnia and Herzegovina). The Bosnian Mine Action Centre (BHMAC) was immediately deployed and provided data and information about the affected regions, the types of influence, the impact intensity, the spatial distribution, as well as the priorities [11]. Obviously, the problem of shifting minefields also hampered the provision of aid and relief, as well as the clearing of debris, as relief workers had to proceed with much care.

Among many other international Search and Rescue (SAR) teams, the Belgian First Aid and Support Team (B-FAST), which is an ICARUS project partner, was deployed in Bosnia to help with relief operations. In order to put into practice the research efforts performed within the ICARUS project, the Belgian Royal Military Academy decided to send along with the B-FAST team a UAV and a trained operator, together with 3D mapping tools, in order to assist the teams with tasks such as damage assessment, situational awareness, dike breach detection, mapping, aerial inspection and relocalizing the many ERWs that were displaced due to the landslides [12]. The computing and data management tools described in Chapters 6, 8 and 9 were used in order to give the end users access to the data gathered by the unmanned system.

This mission fitted perfectly in the framework of the European research projects ICARUS [7] and TIRAMISU (on humanitarian demining) [13]. On the terrain, we were deployed in assistance to a team of the Bosnian Mine Action Centre in multiple regions of the country in order to localize the displaced ERWs. In a first phase, we provided support for urgent actions (urgent demining, assessment of status of minefields, etc.) by performing aerial surveys.

As we wanted to be fully incorporated into the deployment of the multiple international rescue teams, we first had a coordination meeting in the capital city of Sarajevo with the Bosnian Ministry of Security. After discussion with the Ministry of Security of Bosnia and Herzegovina and the national Directorate of Civil Aviation (BHDCA), flight permits for our operations were granted up to a flight altitude ceiling at 150 m for the complete Bosnian territory. Due to the crisis, and because all application documents for the flight permits were readily available (as they were already prepared for the previous operations), these flight permits were issued within half a day.

During a 2-week period, we deployed a vertical take-off and landing remotely piloted aircraft system at 13 locations in the north and central part of the country. In total, we performed about 20 flights within visual line of sight in semi-urban and urban. Two types of operations were performed:

- **Manual flights**, where end users (demining or rescue teams) indicated interest points they wanted to see investigated by the UAV, mainly for damage assessment and visual inspection. A trained operator executed the flights themselves.

Figure 28. Top left: Flooded city of Orasje; top right: UAV used for the operations; bottom left: optimization of the location for the B-FAST water pumps and bottom right: broken dam on the Sava river detected by the UAV (source: ICARUS).

- **Waypoint-based mapping flights**, where an area to be mapped was indicated by the end users. A flight plan for the UAV was then set up to map this area using an autonomous waypoint-based flight. Also for these operations, a trained pilot always supervised the remote control station.

A typical flight lasted around 25 to 30 min, enabling us to cover an area of about 1 hectare. Multiple mapping missions were performed, gathering from 200 images to a maximum of 500 images, all with a resolution of 24 megapixels and mapping areas as large as 1 km^2.

4.2. Flood relief operations

The Belgian B-FAST team was deployed to the city of Orasje (located in the northeast), which was one of the cities that was hit most by the floods. The UAV was first deployed here to assist the B-FAST team to monitor the water levels and to assess the optimal location to install the high-pressure pumps, as shown in **Figure 28**. The problem with the installation of the water pumping system was that water levels were not decreasing after multiple days of pumping, due to an undetected dike breach.

The ICARUS-TIRAMISU UAV was able to locate this broken dam, as shown in **Figure 28**. Expert analysis of the UAV imagery indicated that this dam breach was not caused by natural means. As a result, the Bosnian Ministry of Justice has initiated a criminal investigation against the individual(s) who caused it and commissioned the ICARUS-TIRAMISU UAV image material as evidence. The UAS proved very useful to quickly detect dike breaches and to map the area quickly. One of the main challenges was to find a landings spot on dry land,

Figure 29. Top left: City of Maglaj; top right: re-location of mines due to the landslides; bottom left: detected anti-personnel mine (mine moved due to the landslides); bottom right: damage assessment for mapping infrastructure damage) (source: ICARUS).

as there were virtually no spots of clear and open land suited for take-off and landing. Due to these difficult operating conditions, all take-off and landing operations were done via remote control by a trained pilot.

Next to the operations in support of B-FAST, the UAV was also deployed at the request of the German Federal Agency for Technical Relief (THW) team and Austrian relief workers working at the incident site. These teams asked for the help of our UAV system for damage analysis, aerial inspection, improving their situational awareness and for selecting the optimal location for the installation of the high-pressure water pumps. The Ministry of Security and the Federal Civil Protection of Bosnia and Herzegovina also requested for UAV support in the region of Kopanice (Southeast of Orasje), where the floodwaters from the Sava River broke through the local dams. The floodwaters flowed through the breaches and entirely submerged the farming lands, and all the people needed to be evacuated. The broken dam area was in a mine-suspected region, making this mission especially risky.
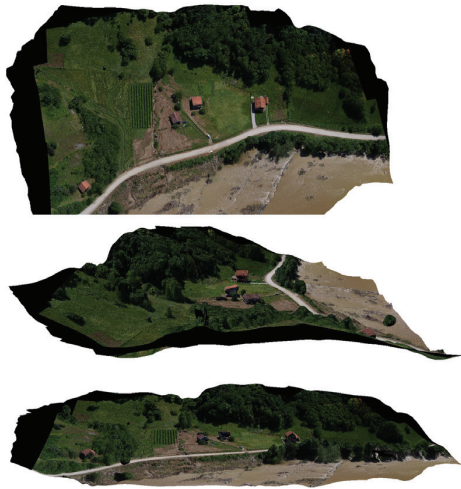

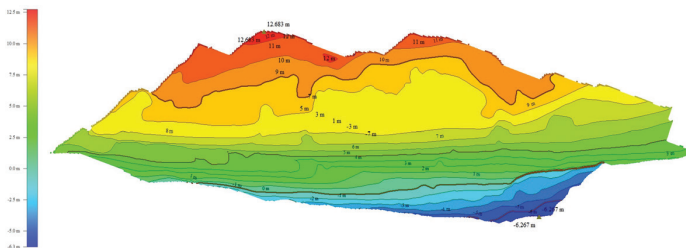**Figure 30.** High-resolution orthomosaic of a mine-affected area (source: ICARUS).

Figure 31. Digital elevation model of a minefield affected by a landslide (source: ICARUS).

4.3. Demining support operations

Another city that was hit hard by the floods and landslides was the city of Maglaj, shown in **Figure 29**. An extra problem in this region was the presence of many ERWs, making the deployment and work of the relief teams very dangerous. Therefore, it was decided to deploy the UAV system for inspection flights, specifically into areas that the relief teams could not easily access due to the mine risks. The UAV was used for aerial assessment and mapping of mine-suspected areas and to find indicators of where the minefields were shifted due to the floods and landslides. **Figure 29** shows such a reallocated minefield due to landslides. The data of the UAV was very important in assessing the ground movement due to landslides, as shown in **Figure 29**. From this information, experts could deduce the area where the land-mines were moved to, which allowed to drastically reduce the search areas.

Connection type	Time (min)	Theoretical upload speed (Mbit/s)
High-speed Ethernet	6	5.5
High-speed Wi-Fi	9	5.38
Mobile connection using mobile phone as router	67	1
Local network of the cloud system (Wi-Fi)	0.79	70

Table 1. Data upload speed for the cloud system (dataset: 500MB).

Figure 30 and **Figure 31** show the initial post-processing results of the Dolac region in central Bosnia and Herzegovina. The UAV was used in this region for providing orthophotos, 3D maps and Digital Elevation Models of the environment to analyse the effects of the landslides on mines and ERWs. These results were used as initial models by BHMAC for spatial estimation of new hazardous risks caused by the movement of unexploded ordnance (UXO) and landmines to areas that were not mine-infested before the disaster. Combining the data from the UAV (3D Digital Terrain Models) with pre-existing data (mine risk maps from the Mine Action Centres and satellite imaging), it was possible for experts to predict the movement of the landmines and to create updated maps of mine-affected areas and mine risk map. To give an indication of the scale of the problem: some mines were found up to 23 km from their original location. If mines can move over such long distances, this means that the search area is enormous and it also means that area reduction techniques, such as the use of UAVs, combined with 3D mapping tools predicting the ERW movements and limiting the search area, have a major impact in the disaster-response operation. More detailed information about this mission can be found in [14].

4.4. Data fusion and sharing

While the UAV tool provided a wealth of 3D information, one issue was that this valuable data had to be distributed quickly and securely to widely dispersed stakeholders (Bosnia and Herzegovina Ministry of Security, demining experts within the Mine Action Centre and deployed international search and rescue teams). An important constraint for doing this was that these stakeholders in general did not dispose of advanced 3D-viewing tools. As a result, we decided to upload the gathered datasets to a cloud system (see Chapter 9). During this field operation,

this was done by directly copying the datasets to the cloud system [15]. We thereby tested the upload speed for a dataset of 500 MB, representing an average size of a dataset acquired by one sensor system (point clouds from a laser scanner or images from the digital camera of the UAS) for a single team area of operation (roughly 200 × 200 m). The results are shown in **Table 1**:

From our point of view, the best configuration is the local network of the cloud system, as it shows limited upload times inside the local cloud created by the server. After uploading and processing the datasets, they are immediately available to the end users for analysis and helping in the decision-making processes. Data processing requires an operator to combine different types of datasets. The operator is also responsible for choosing the correct parameters for the different processing step and also evaluating the results. To facilitate the work of the operator, a set of recommended parameters were provided with a software module. The processed data were provided to the end users via a set of dedicated tools, focusing on efficient data visualization. This entire process is also secure (which is important, as UAV data is in general considered sensitive by governments), as the data itself is not passed directly, but in the form of renders. This also requires also only a very small amount of computation power on the local hardware of the end users. The provided server system is scalable, as it allows easy integration of extra software.

Using this tool, a demining expert from the Bosnia and Herzegovina Mine Action Centre was able to browse through the UAS data sets and make use of his prior experience for identifying indicators of mine presence under the destructive impact of the landslides and floods. He could define new maps of mine-affected areas and create updated mine risk maps. This new information of the mine action situation could be easily shared via the cloud with other end users (mainly SAR teams in this case) in order to enhance their safety on the terrain.

4.5. Conclusions of the flood response operation

Overall, the UAV mission proved to be a real success, based on the feedback provided by the end users. Some of their comments are as follows:

> *The first flight with the UAV was an assessment inspection of the flooding area where the B-FAST water pumps were prepared. To get relevant information of this area from water and land would have cost us about 3 days. With the UAV we were able to provide even better results within 2 hours. With the second flight, we obtained a 3D model of the area on which the B-FAST could determine the natural flow of the water. This information was not even attainable from the ground. (B-FAST Team leader).*

> *The results obtained by the UAV have been of the utmost importance during the response period, and also for post processing and investigation of future activities. (High ranked representative of the Ministry of Security Bosnia and Herzegovina).*

> *The rapid mapping activities and the results we obtained from the UAV mission were crucial for damage assessment, and for relocalizing the many explosive remnants of war that were displaced due to the landslides and flooding. In that situation, we did not risk putting humans in the danger zones. (Technical operation officer of BHMAC).*

This valuable feedback from the end users during a real relief mission clearly shows that the mission really had an impact on the terrain and that the use of novel technological tools developed within the ICARUS project in real search and rescue missions provides an added value.

5. General conclusions and acknowledgements

This concluding chapter has shown how the unmanned tools for search and rescue, developed within the framework of the ICARUS project, have been operationally validated during large-scale simulated demonstrations and even during real-life interventions. A main focus point during all these validation events was a very tight integration of the ICARUS tools into the operational toolbox of real search and rescue workers and the integration into their standard operating procedures, in order to not only validate the technical capabilities of the robotic systems which were developed, but also validate the deployment, command and control, training and support tools which were developed for the end users.

The validation was performed by putting the ICARUS tools in the hands of end users and letting them learn to use the systems and letting them evaluate their experiences. The outcome of all these validation trials was extremely positive. Testimony to this statement is that multiple ICARUS end users have decided—after witnessing the positive effects of the use of ICARUS tools on the terrain—to start the acquisition of their own unmanned systems, and then mostly unmanned aerial vehicles that show the most short-term potential for improving the effectiveness of search and rescue operations. Of course, we must not be blind to the still-existing bottlenecks for the integration of unmanned SAR tools in real rescue operations. For aerial systems, access to airspace is subject to national legislation which is very different from one country to another, which can seriously restrict the use of these systems. For terrestrial systems, mobility on rough terrain and stair-climbing ability are still unsolved issues. Likewise, the unmanned maritime systems still have to prove their effectiveness in very rough sea conditions.

All the presented validation trials would not have been possible without the kind support of many end-user organisations which contributed to the definition of the validation protocols. Specifically, we would like to thank the Portuguese Navy and Belgian Defence for the support received during the marine and land demonstration events. We also wholeheartedly want to thank the Belgian First Aid and Support Team and the Bosnian Mine Action Centre who took up the challenge to allow the integration of novel technological tools developed within a research project inside the real relief mission for the Bosnian floods. This must have been a gamble with uncertain outcome for them at the time of the decision, but it turned out extremely positive for all partners and led to a better mutual understanding between researchers/platform developers and the end-user community. This tight intertwining of the research community and the end-user community has been the main focus of the ICARUS project, such that solutions could be developed which make a real difference on the terrain. Within this chapter, we have clearly shown that such an impact could be made for all of the ICARUS tools discussed within Chapters 3–9.

Acknowledgements

Author details

Geert De Cubber[1]*, Daniela Doroftei[1], Haris Balta[1], Anibal Matos[2], Eduardo Silva[3], Daniel Serrano[4], Shashank Govindaraj[5], Rui Roda[6], Victor Lobo[7], Mário Marques[7] and Rene Wagemans[8]

*Address all correspondence to: geert.decubber@rma.ac.be

1 Royal Military Academy of Belgium, Av. De La Renaissance, Brussels, Belgium

2 INESC TEC - Institute for Systems and Computer Engineering, Technology and Science and FEUP - School of Engineering, University of Porto, Porto, Portugal

3 INESC TEC - Institute for Systems and Computer Engineering, Technology and Science and ISEP - School of Engineering, Polytechnic of Porto, Porto, Portugal

4 Eurecat Technology Center, Av. Universitat Autònoma, Cerdanyola del Vallès, Barcelona, Spain

5 Space Applications Services NV/SA, Leuvensesteenweg, Zaventem, Belgium

6 ESRI Portugal, Rua Julieta Ferrao 10, Lisboa, Portugal

7 Escola Naval, Rua Base Naval de Lisboa, Almada, Portugal

8 Belgian First Aid and Support Team, Karmelietenstraat, Brussels, Belgium

References

[1] Doroftei D, Matos A, De Cubber G. Designing Search and Rescue Robots towards Realistic User Requirements. Applied Mechanics and Materials. 2014;**658**:612-617. Iasi, Romania

[2] Doroftei D, Matos A, Silva E, Lobo V, Wagemans R, De Cubber G. Operational validation of robots for risky environments. In: 8th IARP Workshop on Robotics for Risky Environments; January 2015; Lisbon. Portugal: IARP; 2015

[3] Jacoff A, Messina E, Huang HM, Virts A, Downs A, Scrapper C, Norcross R, Schwertfeger S, Sheh R. Evaluating mobile robots using standard test methods and robot performance data. Technical Report ed. National Institute of Standards and Technology; 2012

[4] Ackerman E, Guizzo E. DARPA robotics challenge: Amazing Moments, Lessons Learned, and What's Next; IEEE Spectrum. 2015

[5] Marques MM, Parreira R, Lobo V, Martins A, Matos A, Cruz N, Almeida JM, Alves J.C, Silva E, Będkowski J, Majek K, Pełka M, Musialik P, Ferreira H, Dias A, Ferreira B, Amaral G, Figueiredo A, Almeida R, Silva F, Serrano D, Moreno G, De Cubber G, Balta H, Beglerović H, Govindaraj S, Sanchez JM, Tosa M. Use of multi-domain robots in search and rescue operations – Contributions of the ICARUS team to the euRathlon 2015 challenge. In: IEEE OCEANS; April 10-13, 2016; Shanghai, China: IEEE; 2016

[6]   Schneider F.E., Wildermuth D., Wolf H. ELROB and EURATHLON: Improving search & rescue robotics through real-world robot competitions. In: 10th International Workshop on Robot Motion and Control (RoMoCo); Poznan University of Technology, Poznan, Poland. 6-8 July 2015;2015. DOI: https://doi.org/10.1109/RoMoCo.2015.7219722

[7]   De Cubber G, Doroftei D, Serrano D, Chintamani K, Sabino R, Ourevitch S. The EU-ICARUS project: developing assistive robotic tools for search and rescue operations. In: IEEE International Symposium on Safety, Security, and Rescue Robotics (SSRR); Institute of Electrical and Electronics Engineers (IEEE), Linkoping, Sweden. October 2013; IEEE; New York, USA. 2013

[8]   Marques MM, Martins A, Matos A, Cruz N, Almeida JM, Alves JC, Lobo V, Silva E. REX14 – Robotic exercises 2014 – Multi-robot field trials. In: MTS/Institute of Electrical and Electronics Engineers (IEEE) OCEANS 2015; Washington, USA: February 8, 2016 2015. pp. 1-6

[9]   Govindaraj S, Chintamani K, Gancet J, Letier P, Van Lierde B, Nevatia Y, De Cubber G, Serrano D, Bedkowski J, Armbrust C, Sanchez J, Coelho A, Palomares ME, Orbe I. The ICARUS project - command, control and intelligence (C2I). In: Safety, Security and Rescue Robots; October 2013; Sweden. Institute of Electrical and Electronics Engineers (IEEE); New York, USA. 2013

[10] Assessment Capacities Project (ACAPS). Floods in Serbia, Bosnia and Herzegovina and Croatia [Briefing Note]. France, Switzerland. 2014

[11] Orahovac A. Mine action after the floods regional synergy in emergency response, technology development and capacity building. In: 22nd Organization for Security and Co-operation in Europe (OSCE) Economic and Environmental Forum; September 12, Wallnerstrasse, Vienna, Austria 2014

[12] De Cubber G, Balta H, Doroftei D, Baudoin Y. UAS deployment and data processing during the Balkans flooding. In: Institute of Electrical and Electronics Engineers (IEEE) International Symposium on Safety, Security, and Rescue Robotics; New York, USA. 2014

[13] Yvinec Y, Baudoin Y, De Cubber G, Armada M, Marques L, Desaulniers JM, Bajic M. TIRAMISU: FP7-Project for an integrated toolbox in Humanitarian Demining. In: Geneva International Centre for Humanitarian Demining (GICHD) Technology Workshop; Chemin Eugène-Rigot, Geneva, Switzerland: 2012

[14] Balta H., De Cubber G., Baudoin Y, Doroftei D. UAS deployment and data processing during the Balkans flooding with the support to mine action. In: 8th IARP International Workshop on Robotics for Risky Environments; January; Lisbon. Portugal: IARP; 2015

[15] Balta H, Bedkowski J, Govindaraj S, Majek K, Musialik P, Serrano D, Alexis K, Siegwart R, De Cubber G. Integrated Data Management for a Fleet of Search-and-rescue Robots. Journal of Field Robotics. Wiley. Hoboken, NJ, USA. 2016;**34**(3):539-582. DOI: 10.1002/rob.21651