GCSs now come in a range of form factors, with their screens, physical controls, radios, processors and software all optimised for each user's exact requirements (Courtesy of UXV Technologies)

Rory Jackson looks at how the key role of a modern GCS is shifting from actual control to planning and monitoring missions

Now at the planning stage

As of 2021, the term 'GCS' (ground control station) has become something of a misnomer for the computer systems used to operate unmanned vehicles. Major advances in autonomy mean they are used less and less for actual control of unmanned systems: most operations now follow pre-planned waypoints and course corrections with very limited interventions.

GCSs are therefore more important for planning missions or monitoring their progress, rather than any actual control, albeit with the operator occasionally having to intervene owing to unforeseen events.

Also, calling them 'stations' might imply that they are large, fixed constructions such as those traditionally used for operating remotely piloted aircraft, but contemporary GCSs can range from small laptops to handheld tablet-type devices.

As a result, GCSs no longer need to be operated from the ground: an operator can walk around a ship deck, be driven around in a car or lean out of a helicopter as they track an autonomous system's mission.

As GCS hardware is increasingly designed for compliance with standardised comms protocols and air traffic solutions, and software systems adapt the latest safety, AI and cloud technologies, it is worth investigating the current array of mission management systems to understand the benefits they can provide to unmanned systems operators.

Hardware designs

Computer hardware is now so modular and powerful that GCS platforms can be built in whatever form factor makes sense for a given mission application, in terms of ergonomics, cost-effectiveness, lightness and environmental proofing or ruggedisation.

For example, there are numerous tablet-type systems with screens as small as 7-10 in across for end-users who want the maximum flexibility with when and where their missions take place, and they are the lightest interface platforms, barring smartphone-based control software, which is still relatively rare. Smartphones also lack the hardware configurability of tablets, and can cause mis-clicks and other input errors owing to their small size.

For emergencies such as monitoring wildfires, routine fieldwork such as corridor mapping of power lines or a USV carrying out a hydrographic survey, the use of these handheld systems are key to minimising the amount of equipment and thus prep time associated with 'unmanned' missions.

These systems are often built around Panasonic tablets such as the FZ-B2. These and many other COTS tablets are well-suited as a platform for whatever combination of GCS hardware is required, as well as core mission software and plug-ins.

Similar but industrial-grade laptops and desktop computers are used to build larger, single- or dual-screen GCSs for use on an operator's lap or desk, be it for field missions requiring more computing power or for missions that routinely take place near a user's offices.

Naturally, these can contain far larger and more robust processors as well as greater quantities of RAM, bigger power banks and more powerful radios. These can enable not only complex visualisations and analytics of real-time data in the field, but also advanced AI capabilities such as swarm control and more.

To ensure sufficient processing and storage bandwidth for software applications and data, GCSs these days will often use the latest generations of dual-core Intel CPUs, GPUs, SSDs, and wi-fi and Bluetooth receivers. The latter is potentially very useful for receiving remote ID data from nearby UAVs.

Application software is often visualised and configurable through a touchscreen, and high-end GCS manufacturers will offer anti-reflective or anti-glare treatments to counter bright sunlight. These treatments might be coatings, custom-designed screen protectors or something more complex such as optical bonding. This last one is a process in which the multiple layers of a touchscreen are vacuumed together with an epoxy bonding medium to prevent reflections or glare occurring between the layers.

A particularly high-end GCS might combine all three. Also, the screens of computers and tablets used as GCS cores often have brightness levels twice that of consumer devices, and potentially five times the brightness for ruggedised case-type GCSs.

Around these screens and core electronics, it remains common for unmanned systems OEMs to customise these base platforms with joysticks and buttons for physical control (in addition to the QWERTY keyboards that come pre-packaged with laptop- or desktop-based systems). This is critical, as missions are increasingly conducted from small boats, armoured vehicles and other moving platforms where turbulence can make it extremely unsafe to try to operate a touchscreen with any finesse.

Joysticks for example are rarely used for flight control these days, except as back-ups in case of autopilot failure, but they are prized for their occasional remote control of camera payloads, particularly if a UAV's embedded vision software recognises something worth a closer look.

Single buttons can be programmed with macros for running lists of launch sequences, or pre-flight and post-flight system checks, reducing much of the operator's workload to a mere tap of a button here and there. Also, both buttons and joysticks can be protected with brackets to prevent accidental commands if the tablet is dropped and lands joystick-first.

These days, tablet GCSs increasingly come with intelligent software-defined radios and antennas, to enable

▶

Ground-based tracking antennas can be installed where needed for maintaining high-fidelity data streams from autonomous vehicles (Courtesy of Embention)

Some ground comms stations are being designed for permanent installation on buildings, providing persistent comms to routine survey or delivery UAVs (Courtesy of Aerobits)

continued connectivity while their users move around with them.

As with their physical control interfaces, these are custom-designed to ensure a modicum of consistency in their wave propagation and latency. While UAV autonomy is now robust enough that few commercial UASs truly need persistent control links from a practical or safety standpoint, providing a GCS with a poor data link could be considered a breach of regulations that require an operator in the loop at all times.

It is common now for high-end GCS manufacturers to offer an initial development kit as the first part of a back-and-forth process to deduce the precise form factor, interface design and ancillary systems needed to produce a hardware platform that works and feels exactly how an unmanned systems manufacturer or operator requires.

That process can take anywhere from a week to a year, but should cover the types of controls, radios, antennas, operating system and ruggedisation that suits each end-user, as well as the amount of RAM, hard disc space and number of I/Os they require.

Well-known standards such as Mil-Std 810G are often adhered to, but GCS development processes these days can also unearth less well-known but highly useful design standards. For example, IEC 60945 provides requirements on EMC for navigational aids in the maritime space, which can be particularly stringent with respect to the selection of radios – one GCS designed to this standard has made innovative use of a Power-over-Ethernet cable in order to meet its requirements.

Data links

As mentioned, various smart radios and advanced antenna systems are available for optimising the connectivity between small tablet GCSs and unmanned vehicles. However, when running high-stakes missions that require constant visibility of the location, health status and survey data stream of a vehicle, using a data link system with a direct line of sight to the unmanned vehicle becomes imperative.

For such missions, it can pay dividends to install a dedicated tracking antenna and radio system in an open space or atop a building with a clear view towards the unmanned system's operating location. The antenna or radio can then connect with the GCS via an extended USB or Ethernet cable – or even remotely using wi-fi – to ensure as high-fidelity a link as possible.

High-end products designed with this use-case in mind have various advantages over smaller data links packaged with the GCS. The most straightforward is that the antenna can be elevated flexibly using telescopic or swappable masts, improving the range and quality of connection and reducing interference caused from the ground and surrounding structures and devices.

Also, decoupling the data link from the packaging of a GCS allows different comms technologies to be selected individually to suit the end-user's budget and mission requirements.

For example, there is a wide variety of antenna designs available. There are now sector antennas that can track ADS-B signals from 500 km away (or FLARM traffic awareness up to 70 km away) for a comprehensive unmanned traffic management (UTM) capability, and while such systems are directional antennas that tend to operate on beam widths of 60-120°, there is no reason why an end-user cannot customise a solution composed of three to six sector antennas arranged radially for 360° coverage, if needed.

Alternatively, a system could be designed with embedded control actuators and precision encoders to enable an antenna to autonomously point itself as needed to maintain coverage of an unmanned vehicle based on received location and heading information. With automated tracking, any type of directional antenna could be incorporated to match the operator's familiarity or budget.

As with the unmanned vehicle itself, several key subsystems would be needed to estimate the position and attitude of the tracking antenna, such as

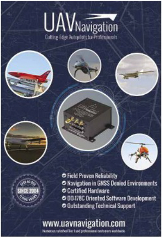
It is common now for high-end GCS manufacturers to offer a development kit as the first part of a back-and-forth process to produce a hardware platform that works and feels exactly as the operator requires

one or two GNSSs (potentially with RTK processing), a triple-axis IMU, barometers and possibly a magnetometer. While that would inevitably drive up mission costs, subsystems can again be selected on an individual basis to derive cost-to-performance improvements across the range of navigation sensors available.

Lastly, newer data link technologies are rapidly being adopted to improve the capabilities of the ground infrastructure for unmanned vehicles. For example, if and when autonomous vehicle operations become routine, LTE connections between office-based GCSs and remotely placed tracking antenna and radio systems can enable persistent, worldwide access to data streams of air traffic or payload data.

New Bluetooth technologies are also enabling ground-based tracking of UAVs' remote ID from more than 3 km away. The development of highly sensitive receivers and high-gain antennas have been vital to this, as FCC regulations limit the output power of remote ID broadcasts to 10 dBm, which has typically been enough for only about 1 km using conventional receiver technology.

Incorporating these new innovations into GCS systems can allow better organisational usage of the survey data coming in from unmanned systems, as well as real-time UTM by different traffic management authorities for long-term safety. ▶



New GCS software is enabling control and monitoring of large numbers of unmanned systems from multiple different devices simultaneously (Courtesy of Autonodyne)

Relying on such remote infrastructure for overseeing UAV missions might bring up additional concerns over equipment integrity though, and raise the peculiar question of whether more UAVs are needed in order to inspect the UAV infrastructure. However, modular installations of companion computers and sensors can be used to monitor the key health parameters of radio stations, identifying precisely when maintenance is needed, as with unmanned vehicles and engines now.

Mission AI

A distinction must be made between GCSs as hardware and as software, as programs written for monitoring and controlling unmanned systems are increasingly agnostic regarding the type of platform they can be installed on.

There are a few obvious features shared by GCS software systems though, such as a top-down map view, with displays and overlays of mission data on vehicle locations, survey information and air traffic feeds. Also, it is increasingly common for GCS developers to use modern gaming engines such as Unity or Unreal for high-quality visualisations of mission environments.

However, the latest GCS software products are advancing the capabilities of operators beyond this standard data visibility in several key ways. One of these comes from designing software that enables autonomous management of multiple vehicles at once across air, land and sea, which can be vital to efforts in disaster relief, defence or wildfire monitoring and mitigation, for example.

Interfacing across multiple different vehicles and control devices is accomplished reasonably easily when coding in C#. A flexible programming language such as this can enable comms across a dozen different protocols, data links or operating systems. It is also widely popular for other reasons – for example, it integrates directly with Unity and other gaming engines that were written in C#, and its similarity to C++ makes it easy for many programmers to learn.

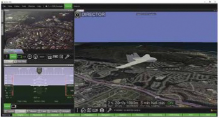
Often, a software module must be installed on each vehicle for translating from its comms protocol, which is often proprietary, into one the GCS can understand. For commercial UAVs, MAVLink has become a popular networking protocol and a standard one for GCS compatibility; NATO's STANAG 4586 is similarly important in the defence world for shared command and control of unmanned systems.

The trickier task is to reduce the cognitive burden on an operator during multi-vehicle missions. Flying a single UAV autonomously over a route of waypoints is simple enough, but the more unmanned assets an operator is managing, the more likely that periodic intervention will be needed owing to onboard fault detections, potential traffic conflicts, surveys turning up points of critical interest that bear closer investigation, and more.

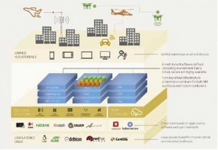
An intricate software engine composed of many pre-written autonomous behaviours is therefore key to providing a human-machine interface that saves the operator having to spin more intellectual plates than is possible without making a mistake. Supplying this successfully means operators can merely click a button in real time or pre-plan conditional autonomous responses based on mission contexts.

These responses could include having one controlled vehicle follow another, or alternatively follow an uncontrolled target vehicle or the GCS (if mounted in a ship or car for example) at a specified distance, perimeter or attitude. More advanced behaviours could include rapidly building and starting a survey flight pattern for a nearby field or building, or engaging in any number of diversionary, offensive or defensive manoeuvres to help accomplish some high-level mission objective.

Also, multiple vehicles could work together. For example, a UAV could land on a UGV or USV to recharge



High-end networking protocols and graphics engines are a must-have for professional operators across defence and commerce (Courtesy of Robot Aviation)



A robust architecture of edge and cloud computing is key to remotely running mission profiles, AI behaviours and analytics (Courtesy of Aquiline Drones)

power and avoid aerial detection while the latter transports it to a new launch location. Alternatively, multiple UAVs could navigate to a new area in a morphing swarm formation, bunching closely together to avoid topographic obstructions or unwanted surveillance, before spreading out again to engage in wide-area gathering of data.

However, making such a solution work is easier said than done. Accurately recognising and gauging data inputs, then correctly selecting and smoothly transitioning into the right sequence of response behaviours, relies on a considerable range of AI technologies.

These include developing semantic reasoning engines that can use sensor inputs to correctly and logically infer the consequences of sensor data, as well as training the sequences of behaviours through various forms of machine learning such as reinforcement learning, unsupervised learning and supervised learning.

However, following this path is key to GCSs unlocking advanced capabilities such as dynamic geo-fencing, seamless handing over the control of more than one or several unmanned vehicles from one operator to another, or even enabling decentralised ad hoc swarming behaviours that have long been touted in academic literature.

AI systems are also key to effectively tracking the hours logged by each component on an unmanned system. Automated alerts and reports can then be supplied to an end-user when something like a muffler, fuel pump or battery is due for replacement.

Tracking component usage can also be used to view the speeds, power outputs or hours of use on vehicles, engines and other subsystems, and thereby visualise trends in how hard or how long operators are running their unmanned systems. By incorporating AI-based maintenance logging into their GCS packages, OEMs can not only ensure operators comply with airworthiness standards on maintenance, they can also understand better how to tailor their next-generation systems to how their customers need them to perform.

Computing advances

Controlling swarms of different unmanned vehicles from a single GCS, and performing other similarly advanced capabilities, requires considerable computer processing power, and UAVs tend to hit their limits in this respect before USVs and UGVs, simply by being more weight-constrained.

A large UAV or UGV equipped with a fuel tank and hybrid-electric powertrain should have little trouble running the latest Nvidia or Qualcomm SoCs, which were developed with computer vision, machine learning and other AI technologies in mind. By contrast, an electric quadcopter small enough to be handheld is far less likely to be able to carry such a powerful computing system, but there are workarounds for this. ▶

Collaborative planning and customisation of GCS interfaces is critical to being able to tailor information displays and alerts for each mission (Courtesy of UAV Navigation)

The most obvious solution for a swarm composed of many different sizes of unmanned vehicle is to use edge computing, with mission analytics and AI algorithms being run on the GCS and using incoming telemetry from the smaller UAVs. The resulting command outputs can then be sent to the smaller craft without needing them to run anything computationally intensive.

In the future, the edge computing approach could be taken a step further, towards collaborative networked operations. In this instance, the most powerful computers on board the largest unmanned vehicles in a swarm could all receive real-time telemetry from the other vehicles.

Those vehicles would then handle all the necessary computational 'heavy lifting', broadcasting control outputs to their subordinate craft.

Formations, manoeuvres and other tactical adjustments would therefore be calculated ad hoc within the swarm. By this time, the GCS will probably be used for nothing more than mission pre-planning and post-mission data analytics; having an operator on-the-loop will only be necessary for regulatory compliance.

Of course, cloud computing is already available for remotely running mission profiles, AI behaviours and post-processing and analytics. With a cloud-based mission management dashboard and data link, any platforms – including tablets and smartphones with limited hard disc space and processing power – can be used for deploying and controlling unmanned systems.

Moving processing duties to the cloud also opens up the possibility of providing data analytics services from there. That would decouple organisations' abilities to exploit their survey data or perform predictive maintenance fixes on their vehicles from their computer hardware or software programming capabilities.

Tailoring a GCS to access cloud-based services potentially means being able to cherry-pick specialised autonomy or analytical features. For example, agricultural survey UAVs could count fruit, track livestock or carry out hyperspectral analyses. Similarly advanced techniques are increasingly available for energy and utilities inspections, first responders missions and other key UAS markets.

In addition to adopting different cloud- or drive-based features as plug-ins for GCS software, programmers also offer increasingly broad ways for end-users to configure their GCS interfaces as they like.

At the moment, this largely means straightforward adjustments such as moving menus or data bars around, or changing the colour coding or appearances of alerts, vehicle icons, geo-fences and so on.

However, future advances in system interoperability and compatibility can be expected to make GCS software compatible with augmented or virtual reality headsets. At that point, operators will be able to interact with 3D projections of map topographies that display swarms of unmanned assets, with real-time updates on position, altitude and heading information.

Using such visualisation for a networked common GCS interface would enable several mission operators and analysts equipped with AR/VR headsets to collaborate in planning a mission, with all of them making adjustments to waypoints and autonomous responses to account for different contingencies. The current alternative is having several people trying to squeeze in front of a single computer screen, or having them sit at different screens but adjusting the same GCS planning interface, creating potential for conflict.

After a launch, multiple operators using an AR/VR-based GCS interface could then scrutinise a shared, common operating picture. They would also be able to access different streams of incoming or logged data without interfering with what the others are seeing, all while keeping their hands reasonably free.

Advances in gaming hardware and graphics engines are worth following, as their adoption stands to greatly improve the handling, quality and intuitiveness of GCS systems over the next several years.

Acknowledgements

The author would like to thank Frank Severinsen and Klaus Aude at UXV Technologies, Niklas Nyroth and Paul Holmstedt at Robot Aviation, Ivan Stamatovski at Easy Aerial, Konrad Cioch at Aerobits, Javier Espuch and Elise Nolan at Embention, Barry Alexander at Aquiline Drones, Kraettli Epperson at Vigilant Aerospace Systems, Miguel Angel de Frutos at UAV Navigation, and Steve Jacobson at Autonodyne for their help with researching this article. ▶