Cyber–physical system architecture of autonomous robot ecosystem for industrial asset monitoring☆

Hasan Kivrak [a,b,*], Muhammed Zahid Karakusak [c,d], Simon Watson [b], Barry Lennox [b]

[a] *Northumbria University, Computer and Information Sciences, Newcastle, United Kingdom*
[b] *The University of Manchester, Department of Electrical and Electronic Engineering, Manchester, United Kingdom*
[c] *Department of Electrical and Electronics Engineering, Istanbul Medipol University, Istanbul, Turkey*
[d] *Department of Electronics Technology, Karabuk University, Karabuk, Turkey*

A B S T R A C T

Driven by advancements in Industry 4.0, the Internet of Things (IoT), digital twins (DT), and cyber–physical systems (CPS), there is a growing interest in the digitalizing of asset integrity management. CPS, in particular, is a pivotal technology for the development of intelligent and interconnected systems. The design of a scalable, low-latency communication network with efficient data management is crucial for connecting physical and digital twins in heterogeneous robot fleets. This paper introduces a generalized cyber–physical architecture aimed at governing an autonomous multi-robot ecosystem via a scalable communication network. The objective is to ensure accurate and near-real-time perception of the remote environment by digital twins during robot missions. Our approach integrates techniques such as downsampling, compression, and dynamic bandwidth management to facilitate effective communication and cooperative inspection missions. This allow for efficient bi-directional data exchange between digital and physical twins, thereby enhancing the overall performance of the system.

This study contributes to the ongoing research on the deployment of cyber–physical systems for heterogeneous multi-robot fleets in remote inspection missions. The feasibility of the approach has been demonstrated through simulations in a representative environment. In these experiments, a fleet of robots is used to map an unknown building and generate a common 3D probabilistic voxel-grid map, while evaluating and managing bandwidth requirements. This study represents a step forward towards the practical implementation of continuous remote inspection with multi-robot systems through cyber–physical infrastructure. It offers potential improvements in scalability, interoperability, and performance for industrial asset monitoring.

1. Introduction

The growing interest in developing mobile robots for inspection, maintenance, and repair (IMR) is reflected by the increasing number of studies being published in literature. IMR robots prove to be valuable in hazardous and challenging environments where humans find it difficult to access, such as mines [1,2], in-pipes [3–5], subterranean environments [6,7], and nuclear power plants that contain radioactive materials [8]. These environments often present unknowns and potential dangers, making the IMR task difficult. Nevertheless, the requirement to inspect and maintain on a regular basis ensures the safety and reliability of the assets and the whole facility, preventing problems that will cost human lives or valuable resources [9].

Inspection robots can be an important tool in industrial asset management, particularly in remote and challenging environments. They can help improve the safety of human workers and allow for more efficient and accurate remote operations in hazardous environments inaccessible to humans. Within a nuclear facility, they are used for a variety of tasks such as decontamination [10], welding [11], leakage inspections [12], exploring and assessing radiation [13,14], navigation with radiation awareness [15], and characterization and inspection operations on the ground [16,17], aerial [18,19] as well as underwater [20,21].

One of the most common applications of inspection robots in industrial facilities is the remote inspection of the environment, which can be divided into the two categories of characterization and monitoring.
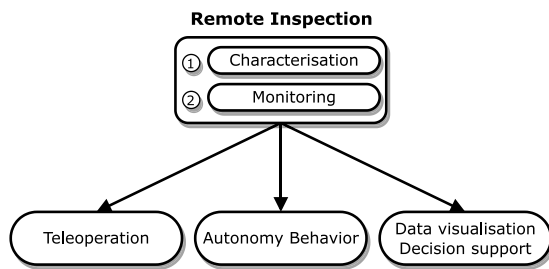
Remote Inspection

Fig. 1. The elements of remote inspection process.

These elements are illustrated in Fig. 1. The characterization can be described as going into the environment which is likely to be unknown and mapping the 2D/3D space or measuring some properties. Once the characterization is done, the monitoring process will involve going back to the same environment and seeing if any changes are occurring by re-measuring those properties. These robots are typically equipped with a wide range of sensors and instruments, including laser sensors, cameras, and mission-specific detectors such as radiation detectors, that allow them to collect data about the facility and identify any potential issues [22]. They can also be controlled remotely by a human operator or operated autonomously, which means they are programmed to navigate and perform tasks on their own [23]. Both ways require getting the data back from the remote system and visualizing it appropriately to help the operator make better decisions, such as changing the schedule of the inspection based on the inspection data they have gotten to date.

Current state-of-the-art inspection robots are typically tested and deployed individually, and their functionality is usually based on the manufacturer's instructions. However, with a single robot, there are limitations in terms of surveying and performing missions that involve complex and comprehensive objectives. The use of multiple robots to perform inspection tasks in a coordinated manner can be particularly useful in large or complex environments where a single robot may not be able to cover all areas or perform all tasks well [24]. Multi-robot systems can also provide a higher level of reliability, efficiency, and effectiveness, particularly in extreme and challenging environments [25]. In the event of the failure of any robot, other robots can be used to compensate for the failed one in order to complete the mission.

Multi-robot systems are proposed for a variety of different cooperative inspection tasks, such as search and rescue [26], underground exploration missions [27], monitoring environmental measurements in green houses [28], and aerial surveys [29]. On the other hand, there are also some challenges associated with employing multi-robot inspection systems, such as the need to coordinate the actions of the multiple robots and ensure that they are working effectively together, as well as the software architecture required for controlling and communicating with robots [30]. The multi-robot systems are further expanded upon when considering the human interaction aspect of robot swarms. The survey paper [31] provides a comprehensive overview of various interaction modalities and emphasizes the importance of considering human factors in the design and operation of robot swarms. This is a crucial aspect that should be taken into account in the field of cyber–physical systems.

Cyber–physical systems enable inspection robots to reach their full potential by connecting them to a digital version of themselves and their environment [32]. This is accomplished through IoT and twinning technologies such as digital twins, in which the real and virtual worlds blend together seamlessly and robots work easily hand-in-hand [33]. There are several potential benefits to using digital twin approaches for inspection robots. One benefit is the ability to test and evaluate the performance of inspection robots in a virtual environment before they are deployed in the field. This can help to identify and address potential issues or challenges that the robot may encounter, improving its reliability and performance [34].

Another benefit of digital twins is the ability to monitor and analyze the data collected by inspection robots, helping to provide a better understanding of the conditions and environment in which the robot is operating [35]. This can be particularly useful for identifying potential problems or opportunities for improvement. Digital twins can also be used to give operators tools to help them make decisions in real-time. The virtual rendering of sensor data can be used to create rich 3D reconstructions of certain environments, which can help reduce operational risks in real-world situations [36].

In the case of multi-robot fleets, the implementation of a cyber–physical system architecture provides better insight into the collective outcome of a mission while keeping track of the real-time status of its individual members. However, digitalization and digital twins also have their own set of potential problems or challenges that need to be considered, such as the communication requirements of seamless integration with physical systems as well as cybersecurity aspects of data security and privacy [37,38].

While significant research interest exists in communication networks, there is still a lack of wireless data standards specifically designed to facilitate information exchange in multi-robot systems [39]. Current network capabilities rely on generic wireless data networking standards that lack the communication-aware algorithms necessary for optimization, especially in addressing challenges such as synchronicity and message frequency [39]. As a result, the development and selection of multi-robot network technologies need to be developed explicitly to manage the dynamics of the application [40].

Teams of researchers through the Defense Advanced Research Projects Agency (DARPA) Subterranean Challenge (SubT) [41] present and investigate the current state-of-the-art methods for operating heterogeneous multi-robot fleets. SubT teams have employed both wired and wireless methods for information exchange between robots and the user interface. Additionally, some teams employed mesh networks to address challenges related to wireless communication networking coverage and accommodate a wide range of bandwidths in underground environments [42]. Despite the unique features adopted by the SubT teams for sharing data, there is convergence and commonality in the approaches used across teams, particularly in preprocessing LIDAR data. This includes relying on local single robot methods such as downsampling (e.g., voxel grid filtering, de-skewing, range-clip filter, intensity-threshold filter) and compression (e.g., draco) to reduce the data volume transmitted to the base station or other robots to manage communication within bandwidth constraints [42]. Furher, Team CoSTAR [43], who won the Urban event of the DARPA SubT, utilized a data management system [44] that uses a practical implementation with the hybrid of ROS 1 and 2 data distribution service over a commercial mesh communication network. Team CERBERUS [45] the winner of the Final event of the DARPA SubT, along with CoSTAR, employed centralized mapping server topologies, where the base station facilitates a collective optimization process across the multi-robot team. In contrast, the remaining teams utilized a decentralized strategy, allowing individual robots to operate independently of one another, with occasional sharing of map fragments. Despite not continuing data sharing, the CTU-CRAS-NORLAB team [46] encountered a 10-second delay in teleoperating robots and frequent crashes in the operator interface. These issues significantly impacted manual control, especially under intermittent networking coverage and varied bandwidths. This highlights the essential need for CPS architecture and emphasizes the necessity for a coordinated optimization process to ensure network integrity and stable data rates. Such optimization involves dynamically adjusting communication capacity while maintaining seamless integration with other system components.

1.1. Contribution

In this paper, our aim is to investigate how a mixed fleet of robots with different locomotion capabilities (wheeled, legged, and aerial)
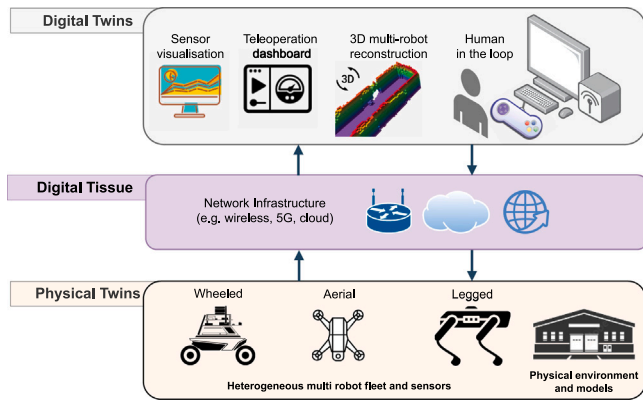
Fig. 2. Conceptual view of the proposed multi-robot cyber–physical infrastructure.

Fig. 3. A UML use-case diagram or higher-level view of the proposed system.

can utilize a CPS architecture integrated with a scalable, low-latency communication network, specifically focusing the characterization of challenging remote environments. This study extends the work in [47] by focusing on the scalability of inspection robots in a heterogeneous fleets of robots. Scalability, here, refers to either increasing the number of robots, which leads to a multi-robot fleet, or dealing with network communication limitations, such as managing network bandwidth and reducing latency.

Our contribution lies in adapting existing solutions to facilitate effective remote inspection tasks within a multi-robot setting. While the use of fleets of robots for inspection and the individual components of the proposed architecture (i.e., physical twin, digital twin, and digital tissue) are not novel in themselves, we have tailored these solutions for the demands of scalable multi-robot inspection such as low-latency communication capabilities that allow for real-time coordination, efficient data management strategies that optimize the use of available bandwidth and adjust to the changing needs of the multi-robot fleet. These features address the unique challenges of coordinating and governing multiple robots in a cyber–physical architecture when compared to the DARPA Sub-T.

Moreover, our work has practical implications for the implementation of continuous remote inspection with multi-robot systems. By demonstrating how these existing solutions can be integrated and adapted to meet the specific requirements of multi-robot inspection tasks, our work is an important step towards putting these kinds of systems to use in a cyber–physical infrastructure.

2. Cyber–physical system architecture for multi-robot fleets

In this study, we propose a new generalized CPS architecture to explore and characterize remote inspection activities in heterogeneous multi-robot fleets. A suite of tools is presented to address the management, scalability, and decision support challenges of remote inspection missions. The proposed system, shown in Fig. 2, consists of three tools: (1) a physical twin, (2) a digital twin, and (3) a digital tissue.

Furthermore, a UML use case diagram, depicted in Fig. 3, is used to identify system requirements through various use cases and interactions coordinated by different types of actors. There are three main actors interacting with the system: the digital twin operator, robots, and researchers. The researchers are responsible for setting up the digital twin interface, choosing a simulated environment, providing and analyzing data, and ensuring that real-world assets are accurately represented in the digital twin interface through communication networks. The robots, equipped with sensors, perform inspections and provide inspection data to the digital twins for visualization and execution of multi-robot mapping use cases. They also receive motor commands from teleoperation data sources such as a real or virtual joystick. Meanwhile, the digital twin operator uses digital twins to control robots, monitor analytic
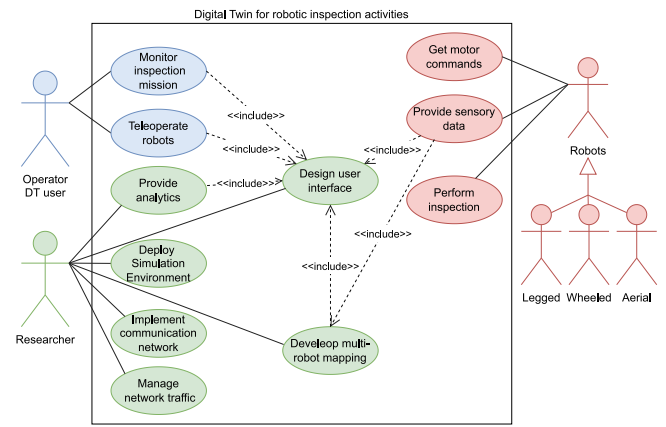
information about real-world assets, such as multi-robot mapping of 3D space, and track state changes of robots to make better decisions based on inspection data.

2.1. Overall communication architecture and integration

In the development of a multi-robot system, the initial step involves identifying the various types of robots and their behaviors in the mission environment to enable autonomous missions. This encompasses essential functionalities such as mapping, localization, navigation, and other foundational capabilities. Secondly, the integration of DT provides overseeing the operations of a multi-robot fleet and enables intervention in inspections when necessary. DT supports real-time sensor visualization through status updates and fosters human collaboration. Finally, the system incorporates features that establish an efficient and scalable communication network for enabling inter-robot coordination and timely sharing of information within the DT interface.

This paper utilizes publicly available digitalization and software tools to present a solution for coordinating a cyber–physical multi-robot ecosystem. ROS is adopted due to its provision of standardized functionalities as an open-source middleware. It provides a flexible framework designed for the development of robot software that can easily scale and be interchanged. Moreover, ROS facilitates the creation of distributed software communication architectures through its publish–subscribe messaging mechanism, operating based on TCP/IP, along with tools for introspection. These capabilities make ROS as a de facto framework for meeting the diverse requirements of multi-robot system architectures.

Fig. 4 provides a graphical summary of these component implementations as well as the data flow strategy within the system of systems and how it is connected to the human operator. The efficient exchange (i.e. reception and transmission) of data in this domain is a crucial aspect of the operability of the cyber–physical system. This makes the integration process a crucial aspect of the current work.

All physical instances of the robots provide updates of its pose and sensor-collected data based on their respective coordinate frames. Data originates from the individual operating systems of each robot and is passed on to the digital tissue. Specifically, the point cloud is first downsampled, then compressed, and the frequency control of the compressed data is maintained by the bandwidth manager, resulting in reduced bandwidth and high transmission performance. After transmission, the decompressed point cloud is then relayed to the digital twin which is then presented to the human operator through a user interface.

The live map, generated by stitching together the sensor data from each robot, is available in the central digital twin node. On the one hand, this centralized approach may limit the autonomy of the individual robots to some extent, as their ability to make globally informed
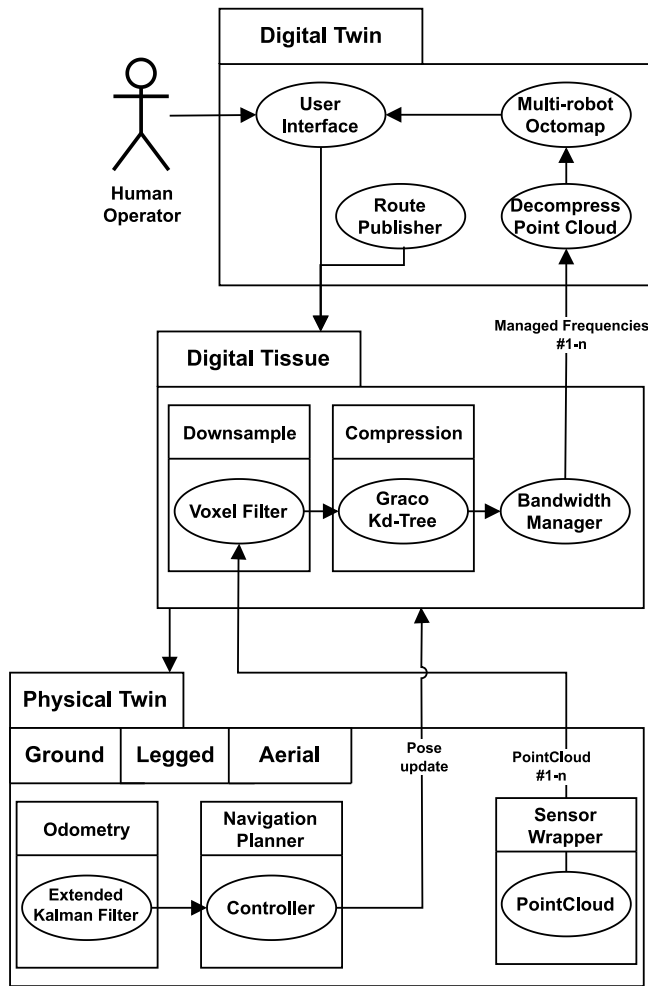
Fig. 4. Proposed software cyber–physical system design and data flow diagram.

Fig. 5. Elements of physical twin.

decisions may be limited without access to the live map generated by the central node. On the other hand, it allows for efficient data management and minimizes the network load. Our system is designed to be flexible to allow each robot to have access to the live map if necessary. This adaptability allows for a balance between autonomy and communication overhead, enabling the robots to benefit from global information while minimizing the network load associated with transmitting map fragments.

In a live teleoperation scenario, the digital twin user interface will be the primary point of contact between the robots and the operators; reflecting the current state of the robots in the physical environment and the coverage of the generated 3D map. This signifies that data accuracy and perceivability must always be preserved while maintaining the lowest latency possible. Finally, the human operator provides its desired navigation and perception commands through the user interface and information is relayed back to the physical instances; closing the loop and attaining bi-directional communication.

Further details regarding the key components of the proposed CPS's overall communication architecture and integration are detailed in the subsequent sections.

2.2. Physical twin

The physical twin represents the real-world physical mission environment, such as the spatial building structure, as well as the physical instances of robots, sensors, and other assets containing useful information from the actual mission environment.
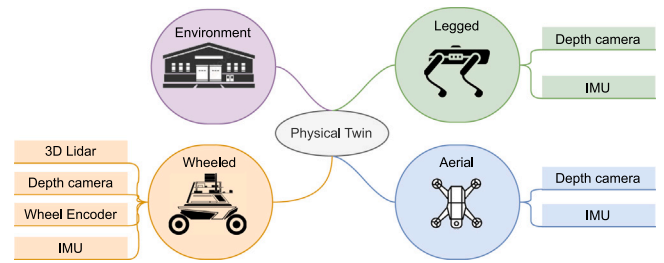
To facilitate effective cooperative inspection missions, it comprises a heterogeneous multi-robot fleet with complementary skills consisting of wheeled, legged robot, and aerial robots, as depicted in Fig. 5, equipped with sensors like 3D Lidars, depth cameras, wheel encoders, and IMUs. These sensors pave the way for robots to enable foundational functionalities such as mapping, localization, and navigation for autonomous inspection capabilities.

2.3. Digital twin

The digital twin is a virtual representation or synthetic environment, synchronized with the real asset and receiving live updates. Digital twins can be used in various ways, such as to develop and test new algorithms quickly and in a manner that simulates the real robot. Utilizing a digital twin in the context of multi-robot inspection can lead to better and more efficient systems by allowing operators to understand the robots' locations and next steps to monitor and map the environment, while also allowing for human interaction through a digital twin interface. This can aid operators in deciding and scheduling inspections based on previous results. In this study, we use the digital twin to build a unified 3D mapping and characterization of the environment utilizing multi-modal sensor data from 3D lidar and visual depth cameras in a complementary manner, and providing remote teleoperation through a virtual joystick with real-time updates of the environment's video streams and 3D maps. Together, they provide a human operator with a complete view and control of the inspection environment, and more information will be provided in the upcoming sections.

2.3.1. Heterogeneous multi-robot 3D mapping

The use of multi-robot systems in various fields has increased the demand for 2D and 3D multi-mapping solutions, which has been extensively studied in literature. Two general approaches are used to enable fleets of robots to create a unified and unique map of the environment: exchanging sensory data or exchanging locally created submaps between robots. However, the map merging or fusion method with unknown initial positions of the robots is a challenging problem and requires a global map integration step to collect and merge the individual maps created by each robot.

Various studies have focused on finding alignments between individual maps for both stitching 2D maps [48,49] and 3D maps [50,51]. Alternatively, scan-matching approaches have been used for multi-robot 2D [52,53] or 3D SLAM [54]. In hazardous and challenging real-life scenarios, it is often possible to know the initial positions of the robots. In this work, we therefore consider a multi-robot 3D octomap by matching the 3D scans through incremental mapping serially for each robot, unlike traditional SLAM approaches. Octomap is a 3D occupancy grid mapping approach [55] that uses a hierarchical data structure called an octree [56] and a geometric modeling method called octree encoding [57]. The proposed architecture for the multi-octomapping algorithm is outlined in Fig. 6. As illustrated in Fig. 6, the system is centralized such that point cloud messages transmitted by multiple robots are processed sequentially using a queue, and voxels are updated
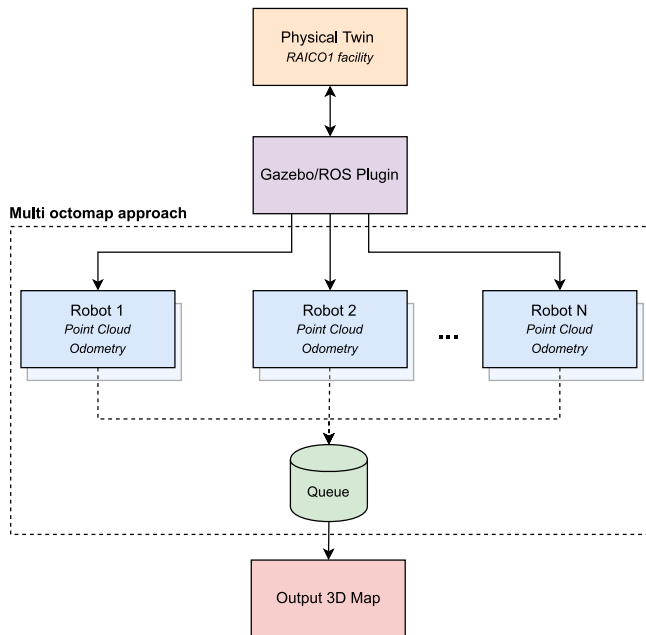
Fig. 6. The proposed architecture of the multi octomapping algorithm.

Fig. 7. Rqt-based user interface interacting with robots during their runtime: (1) live video or camera image frame, (2) Rviz 3D visualization window (e.g., 3D multi map visualization) (3) secondary Rviz window (top view) (4) Logging and status window, (5) remote teleoperation window.

accordingly. The results obtained from the mapping algorithm are then immediately updated in the graphical user interface described in the subsequent section.

2.3.2. Graphical user interface (GUI)

In the context of remote inspections, it is crucial to have viewing systems to monitor and control the tasks. Graphical user interfaces (GUIs) are needed to interact with the digital twin, allowing operators to monitor the status of the entire fleet and execute multi-robot missions. There are various open-source software projects and commercially licensed products available, such as Webviz [58], Rosboard [59], Rosshow [60], Foxglove [61], Formant [62], Rocos [63], and Freedom Robotics [64], that offer web-based dashboards or portals for interactive, multimodal data visualization, control, and debugging of varying capability and scale. However, these tools do not fully meet the specific needs for controlling a robot fleet for efficient multi-robot mission operations.

To address this, we developed an intuitive, and simple Robot Operating System (ROS) Rqt GUI [65] based on the Python binding of the Qt cross-platform framework. Rqt is a plugin development framework that includes a common and robot plugin suite that can be used on or off the robot runtime. It allows for managing all windows on a single screen and turning your own Qt widgets into Rqt plugins with multi-language support. Additionally, while Rqt can communicate directly with ROS messages, these messages need to be formatted by the ROS bridge server package [66] for use in web applications.

In summary, we have designed a unified Rqt-based user interface that seamlessly integrates existing plugins for multi-robot team inspections. The interface features teleoperation tools for controlling the fleet, a multi-modal 3D environment view, real-time logging, and live sensor visualizations from the fleet's visual and geometric sensors. The Rqt user interface, depicted in Fig. 7, is composed of five windows.

1. Video stream window: using the rqt_image_view plugin [67], this window shows all available live video streams provided in a dropdown menu.
2. Primary Rviz window: using the rqt_rviz plugin [68], this window displays the 3D multi-robot mapping of the environment, providing an overview of the fleet and live sensor visualizations,
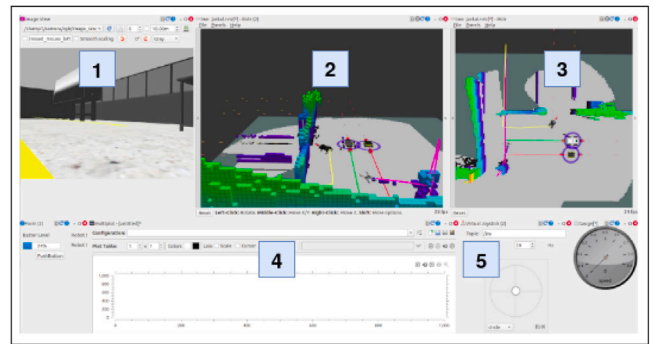
such as point clouds and tf messages, gathered from the fleet members' visual or geometric sensors.
3. Secondary Rviz window: similar to the primary Rviz window, but with a top-view perspective.
4. Logging and status window: using the rqt_multiplot plugin [69], this window provides multiple 2D plots and a status log for fleet health conditions such as acknowledgment, battery status, and connectivity.
5. Remote teleoperation window: this window allows for operating a single robot team member via a virtual joystick, linking each robot command velocity message and providing speed information similar to a car speedometer.

After describing the graphical user interface, the next section of the paper explains how a digital tissue component enables the user interface and the physical environment to exchange data in real-time with efficiency and low latency, ensuring smooth inspection operations.

2.4. Digital tissue

The term "digital tissue" refer to the communication infrastructure and networking of systems in which physical and digital twins are established, also known as cyber–physical fabric [70]. This is yet another critical component of cyber–physical systems that is in charge of efficiently feeding the enormous amount of data generated by the physical twin into digital twins.

One challenge in meeting these requirements is limited communication bandwidth and latency for data transmission, which varies depending on network type as shown in Table 1. The maximum theoretical bandwidth for traditional Wi-Fi ranges from 11 Mbps to 600 Mbps depending on the generation [71], while 4G (LTE) can reach 1 Gbps and 5G can reach 10 Gbps depending on network conditions [72]. Bandwidth capacity can be improved through 4G/5G networks, but there will still be a need for smart bandwidth management due to the bounded bandwidth limitations.

Another challenge is ensuring reliable, resilient, and scalable communication coverage so that we can stay in control of the robot and timely data transmission. In the study, the ROS distributed computing middleware is deployed as a software communication infrastructure to make multi-robot fleets ready for cooperative inspection tasks with a digital twin.

Table 1
Maximum theoretical bandwidths of different network types.

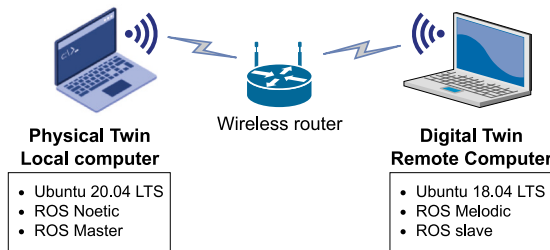| Network type	Bandwidth
Traditional Wi-Fi	11 Mbps to 600 Mbps
4G (LTE)	1 Gbps
5G	10 Gbps

Fig. 8. Wi-Fi communication infrastructure.

2.4.1. Communication infrastructure

As part of the networking infrastructure configuration, Fig. 8 illustrates a wireless local area network (WLAN) without internet connection, to which two devices will be connected. In this scenario, the computers use the standard (802.11) Wi-Fi network provided by a wireless modem router. The available bandwidth on the local network is measured using the iperf network performance tool [73]. The tool operates in server mode on one computer, sending data to the remote computer, which runs in client mode to receive data from the server. Results of the experiments showed that the maximum achievable bandwidth on the local network was 60 Mbps.

Tables 2 and 3 shows the individual and total approximate bandwidth consumption of each robot types, which varies due to the heterogeneous nature of the robots and their different sensors and configurations. While type A refers to wheeled robots equipped with 3D lidar operating at 10 Hz, type B, legged and aerial indicates robots fitted with depth cameras operating at 30 Hz.

The bandwidth requirement of 874.659 MB/s equivalent to 6.997 Gbps (1 MB/s is 0.008Gbps), in the multi-robot fleet ecosystem emphasizes the need for efficient polling, management, and optimization of bandwidth to ensure scalable coexistence during operation. The next section explores scalable and low-latency communication networks through techniques such as downsampling, compression, and bandwidth management for efficient data transmission.

2.4.2. Downsampling point cloud data

As the number of robots in the fleet increases, the load on the communication channel also increases, impacting communication. It is important to have efficient methods for transmitting bandwidth-intensive data types, such as point clouds, RGB, and depth image data, without slowing down the system too much, so that the mission is not affected by message delays or loss. On the other hand, other lightweight message types, such as odometry, IMU, and transformations, have a negligible effect on bandwidth usage.

One way to manage bandwidth consumption is to make smart use of the data by using techniques to reduce the amount of data that is sent. For example, point cloud streams captured from RGB-D sensors have a high resolution (640 × 480) and density of points, which occupy a significant amount of resources. A typical RGB-D point cloud streaming at 30 frames per second (Hz) results in a bandwidth of approximately 2.06 Gbps, equivalent to 640 * 480 pixels * 28 bytes per point (bpp) * 8 bits * 30 Hz, where each point is described with 7 float values (x, y, z, r, g, b, intensity), totaling 28 bytes [74]. Therefore, point cloud streams are commonly filtered [75,76] using methods such as outlier and noise removal, smoothing, and downsampling to reduce the density of the 3D point cloud data.

One common method of downsampling is the voxelized grid approach (voxel grid filtering), which divides the space into a 3D voxel grid (i.e., a set of cubes with a fixed size in space) and retains only the centroids, which are averages of point distribution within the voxels. This ensures a consistent level of detail or uniform resolution across the entire point cloud. The downsampling rate can be controlled by adjusting the voxel size along each dimension. Smaller voxels allow for finer degree downsampling, but this comes at a greater processing cost and a larger output point cloud. Additionally, octree grid filter may also be a way to implement voxel grid filtering method that uses less memory [76].

2.4.3. Compressing point cloud data

When point cloud streams need to be transmitted or saved through channels with limited bandwidth, it becomes crucial to use compression techniques in addition to data minimization techniques. The octree-based [77] and kd-tree-based Google Draco compression schemes [78] offer the ability to compress and decompress 3D point clouds. In this study, we use the Draco-based implementation of the point_cloud_transport package[1], which utilizes better performance for the storage of point clouds [79] and a lossless compression method, as opposed to the lossy octree compression method used by PCL [80].

The task of compressing and decompressing is accomplished by two distinct machines as depicted in Fig. 8. One machine represents the local device or instance of the robot (physical twin), while the other represents the remote device (digital twin). The local system generates point cloud data and publishes a compressed form of it, and the remote machine decompresses the data and republishes it for further processing. This results in faster data transmission and lower network bandwidth usage.

2.4.4. Bandwidth manager

A network traffic monitoring tool, also known as a bandwidth manager, is necessary to ensure that the actual bandwidth requirements match the available bandwidth in the event of an overload. Additionally, it is crucial for systems with multiple robots to share bandwidth effectively to prevent the communication channel from consuming more bandwidth than the system can handle. One way of achieving this is to control the frequency of topics that publish data and send only the data that is needed at certain points in the mission. Therefore, a dynamic bandwidth management strategy is used by optimizing the frequency of topics to deal with the bandwidth bottleneck caused by a large amount of data to be transmitted.

The approach is based on the existing framework in Ref. [81] and involves expressing the bandwidth consumption and frequencies as a linear optimization problem constrained to the available bandwidth as described in Eq. (1). The constraints on the available bandwidth allow for higher priority topics to have a higher frequency remaining within the available bandwidth limit. In other words, the bandwidth manager looks at the current message's length and how often it is being sent and figures out a new frequency rate based on its current priority. The priorities of all robots are initially set to 0. When the robots get to their mission area and the operator and mapping algorithm need more frequent feedback, the priorities are changed to high, which is 1. In the event that the robots finish their routes, the priorities are changed back to 0. By effectively sensing the remote environment using an approach based on environmental events and providing more bandwidth for higher priority messages, the operator can gain near-real-time capability of what is happening in a remote area .

Managed topics are chosen from sensory data which are bandwidth intensive and contribute to the mapping process, such as point clouds generated by 3D lidar and depth camera sensors. 4% of the bandwidth

---

[1] https://github.com/paplhjak/point_cloud_transport

Table 2

Approximate individual bandwidth requirements for physical twin fleet robots in a network based on sensory, joint, and odometry data system settings.

Bandwidth consumption						
Robots	Point cloud data (MB/s)			Joint states + odom (MB/s)		
	min	max	Hz	min	max	Hz
Type A	2.85	4.50	10.0	0.046	0.048	50.0
Type B	280.0	290.0	30.0	0.046	0.048	50.0
Legged	280.0	290.0	30.0	0.046	0.048	40.0
Aerial	280.0	290.0	30.0	0.021	0.023	30.0
Multi-robot fleet	**842.85**	**874.50**		**0.159**	**0.167**	

Table 3

Approximate maximum total bandwidth requirements.

Robots	Bandwidth consumption (max) (MB/s)	Network speed (Gbps)
Type A	4.546	0.0363
Type B	290.046	2.32
Legged	290.046	2.32
Aerial	290.021	2.32
Multi-robot fleet	**874.659**	**6.996**

(a)

(b)

Fig. 9. The physical environment and robots. (a) Robotics and AI Collaboration laboratory (RAICo Labs). (b) The physical instances of robots.

is set aside for messages that are not managed based on the analysis outcomes in [47], like giving tasks and controlling to robots as well as publishing odometry data. That ensures only part of the total bandwidth is used in the calculation topic frequencies.

$$\sum_{i=1}^{n} \text{message\_length}_i \cdot \text{frequency}_i \leq \text{available\_bandwidth} \qquad (1)$$

3. Experimental setup

We designed and conducted experiments to verify and demonstrate the feasibility of the proposed CPS architecture that meets the requirements of heterogeneous multi-robot inspection. In a simulation-based experiment, a fleet of robots will explore or scan the unknown environment and gather data for the purpose of 3D environmental construction to be used in planning future inspection missions.

3.1. Case study

The experiment focuses on a heterogeneous multi-robot inspection mission in a realistic and representative industrial environment. However, due to resource access limitations, we recreated the physical environment and robots in a virtual or simulation environment, replicating the physical twin shown in Fig. 9 as a virtual environment as depicted in Fig. 10. The implementation was carried out using the ROS middleware framework and Gazebo simulation tool on a local area network. The Gazebo simulation environment has been set up with a 3D model of a new robotics and AI collaboration laboratory (RAICo Labs) in Whitehaven. The facility was initiated to help the Nuclear Decommissioning Authority meet its grant challenge [82] by developing robotics and artificial intelligence (AI) technologies. The simulation world model is approximately 20 m by 15 m and 2 m above the ground level. It consists of three separate area fences installed with industrial equipment and tools. Fig. 10 shows the environment simulated in Gazebo and demonstrates the task allocation for cooperative inspection operations using the multi-robot fleets, which each have different locomotion capabilities.

The robot team consists of two wheeled robots, one legged robot, and one aerial robot: the wheeled Agilex Scout2.0 and Clearpath Jackal, the legged Champ robot [83], and a small surveillance micro aerial vehicle called Iris, along with their sensors. Scout 2.0 and Jackal are both four-wheeled, differential-drive ground robots equipped with encoders on their wheels and an IMU for odometry calculation. Scout 2.0 is equipped with a 3D laser with a range of 100 m and a sample size of 1875 @ 10 Hz. Jackal has a depth camera for creating a 3D map of its surroundings. Champ, a quadruped robot, and Iris, a small air vehicle, both have an IMU and a depth camera and are controlled by Lee's hierarchical control algorithm [84] and the Pixhawk flight controller [85], respectively. All the robots have the same depth camera specification, capable of acquiring data at 640×480 @ 30 Hz with a range of 5 cm to 18 m. Tasks are assigned based on their two and three-dimensional navigation capabilities among the fleet of robots and will be overseen and controlled by a dedicated remote operator with shared autonomy.

The digital twin generates multiple routes (a colorful path of arrows as shown in Fig. 14(a)) by sequentially reading from predefined way points stored in separate files labeled as route1, route2, route3, and route4, each determining the path for individual robots. Once the digital twin assigns these routes, the robot team concurrently and autonomously navigates to the next waypoint along their trajectory until reaching their destination. The developed controller uses the current locations of the robots according to the robot's odometry and sends commands to each robot separately to go to the next waypoint in the route. This way, we take advantage of each robot's environmental coverage path with their complementary skills. Furthermore, we use the shared autonomy approach to enable the operator to guide the robot at any time using a robot teleoperation tool (e.g., virtual joystick, visualization) via the user interface by switching incoming velocity commands from the user or controller.

The operator monitors the fleet of robots as it is doing its mission of 3D mapping scenario of an unknown facility using data brought
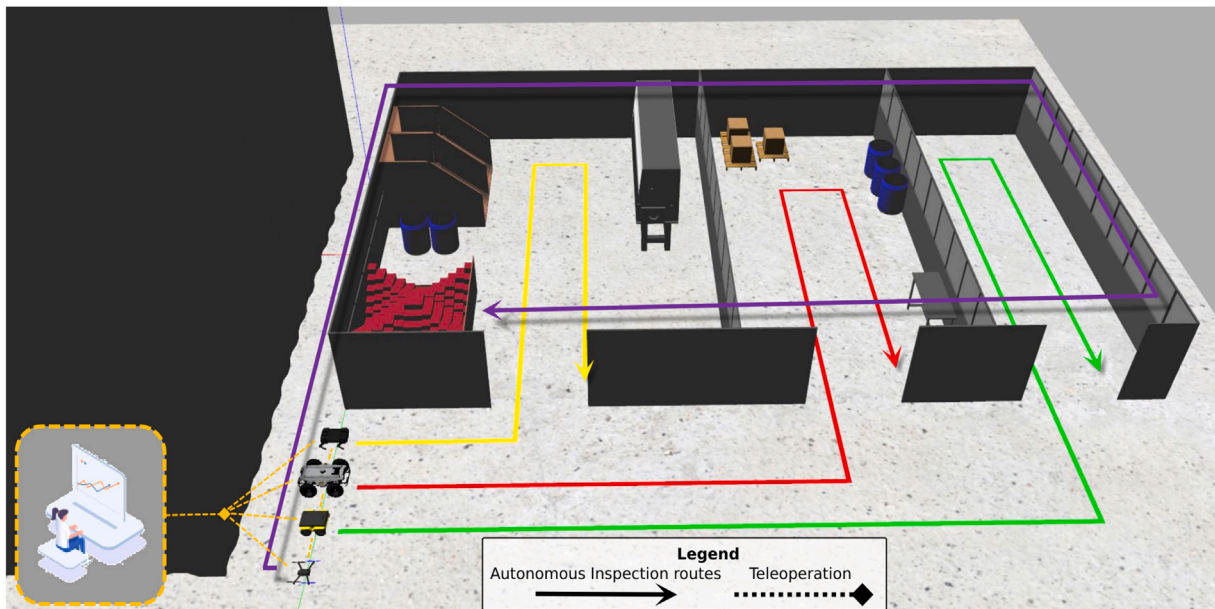
Fig. 10. Start of the mission with each robot in a simulated view of the proposed autonomous multi-robot remote inspection scenario. The target routes are shown in yellow, red, green, and purple, where each robot inspects a separate fenced area.

together from various sensors, robots, and network states, as shown in Fig. 7: (1) a live camera feed, (2) 3D mapping, (3) odometry, (4) lidar data, (5) battery, and (6) bandwidth and latency. The provided information lets the user intuitively operate the robot in an integrated 3D environment while avoiding obstacles when needed.

Additionally, to address resource access limitations, we have replicated the real-world environment in a virtual simulation (see 2.2) so that the physical twin is run on a master node machine with an Intel i7-11800H CPU at 2.30 GHz and 16 GB RAM, using Ubuntu 20.04 and ROS Noetic as shown in Fig. 8. On the remote operator side, the digital twin components, including multi-robot mapping computation and user interface features, are executed on a laptop with an Intel i7-4700HQ@2.40 GHz processor and 16 GB memory, running Ubuntu 18.04 with ROS Melodic, which serves as the follower node.

3.2. Network performance metrics

Network performance metrics are used to evaluate the scalability of the system and the real-time performance of the transfer of data. The following network analysis metrics are used to assess the network performance of the CPS framework.

1. **Bandwidth and publishing rate measurements**:
   Bandwidth and publishing rate, or frequency, are two commonly used metrics in network performance. Frequency, measured in Hz, is the rate of messages, and bandwidth, measured in MB/s, is the potential amount of data that can be transmitted in a network channel [86]. Monitoring both metrics provides a more complete understanding of network resources, allowing for identification of performance bottlenecks and latency. The rostopic command-line tool [87] and iperf bandwidth measurement tool are used to measure frequency, determine the consumed bandwidth, and estimate the total available bandwidth.
2. **Latency measurements**: There are different kinds of delays in a computer network, such as transmission delay, propagation delay, queuing delay, and processing delay. All of these are included in to the computation of the end-to-end delay [88]. The transmission delay, which identifies the amount of time it takes to reach an entire packet through the link. The transmission latency is taken into account for our mobility scenarios, as
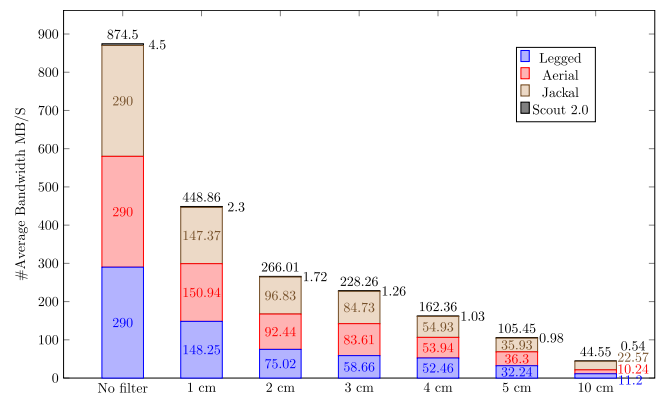

Fig. 11. Downsampling rate of voxel grid filters in relation to consumed bandwidth for varying voxel sizes.

other kinds of delays are not affected by the mobility of the system [89]. It is measured using the rostopic tool to have complete knowledge about message-level transmissions.

4. Results

The proposed CPS architecture was evaluated through realistic simulation experiments, including a series of scalability analyses with multi-robot fleets. Fig. 11 illustrates average amount of consumed bandwidth of multi-robot fleets depending on the different size of voxels. This was a good compromise of reducing required average consumed bandwidth of cyber–physical robot fleets. Smaller voxel size in downsampling typically leads to better preservation of information from the original point cloud. After experimentation, a reasonable voxel size of 10 cm was found for robots, except for Scout 2.0 where 1 cm was found to be more effective in retaining the original point cloud's information.

Fig. 12 illustrates the compression performance of the reduction in required average bandwidth of point cloud streams downsampled at various voxel sizes and compressed using Draco. Table 4 provides a summary of the statistics for processing point cloud data prior to

Table 4

Bandwidth consumptions results of processing point cloud data (voxel size = 10 cm).

	Raw point cloud MB/s			Filtered point cloud MB/s			Compressed point cloud MB/s		
	min	max	Hz	min	max	Hz	min	max	Hz
Scout 2.0	2.85	4.50	10	2.1	2.69	10	0.37	0.46	10
Jackal	280.0	290.0	30	22.14	22.57	30	4.12	4.44	30
Legged	280.0	290.0	30	10.32	12.47	30	1.67	2.47	30
Aerial	280.0	290.0	30	10.53	12.64	30	1.8	2.7	30
Total	842.85	874.50		45.09	50.37		8.01	10.07	

Table 5

Network performance measurements.

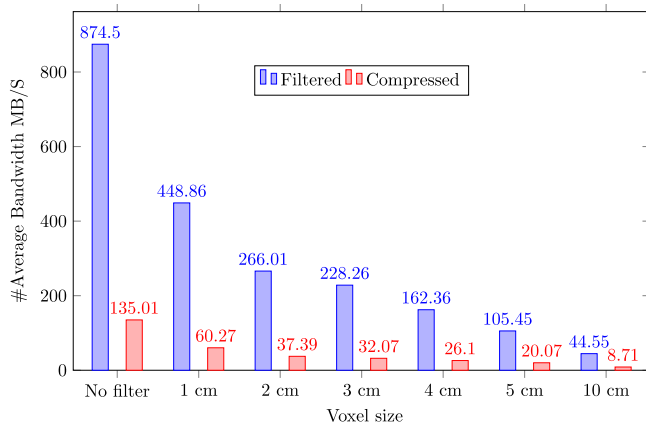|	Consumed bandwidth (MB/s)			Latency (ms)		
	avg	max	std	avg	max	std
Multi-robot fleet scenarios						
Scout	1.195	1.65	0.27	39.0	42.0	28.0
Scout + Jackal	4.70	4.74	0.01	41.0	45.0	28.0
Scout + Jackal + Aerial	5.3	5.67	0.12	43.0	58.0	39.0
Scout + Jackal + Aerial + Legged	4.80	6.57	3.02	52.0	65.0	9.71

Fig. 12. Comparison of filtered and compressed multi-robot point cloud streams.

transmission for each robot. The data is filtered using a 10 cm voxel grid and compressed using Draco.

As seen in Table 4, the bandwidth consumption for each robot varies based on the clutter in the environment and their sensor configurations. These results validate the processing method's efficiency, resulting in a significant reduction in transmitted data.

In addition to processing point clouds, the performance of the bandwidth management algorithm is evaluated. Table 6 displays the frequencies and distribution of bandwidth consumed, aiming at maximizing utilization. The bandwidth manager optimizes consumption by dynamically dividing available bandwidth among robots and assigning frequencies. During the experiment, available bandwidth was around 60 Mbps, and the minimum and maximum frequencies are kept between 1 and 30 Hz.

Table 5 compares the network performance of a demonstration of multi-robot fleet scenarios on different scales with increasing numbers of robots. The bandwidth manager ensures the system stays within bandwidth limits, resulting in latencies of less than 100 ms for each scenario. Finally, Fig. 14 provides a top-down views of heterogeneous multi-robot mapping, showing octree maps of the environment at a resolution of 0.1 m.

5. Discussion

The evaluation of the CPS in a multi-robot fleet showed in Fig. 11 that the bandwidth requirement is reduced from 874.50 MB/s to 44.55 MB/s (94.90%) or by a factor of 19.62 by downsampling the point cloud data with a voxel size of 10 cm. The combination of downsampling and compression resulted in further reduction, leading to a significant decrease in data from 874.50 MB/s to 8.71 MB/s (99%). The results in Fig. 12 allow one to adjust the downsampling rate and/or the use of compression techniques accordingly.

Fig. 13 illustrates how bandwidth consumption increases with the number of robots without the use of bandwidth management for robot types A and B. Type A refers to robots equipped with 3D lidar operating at a maximum data rate of approximately 3.68 Mbps (0.46 MB/s), while type B indicates robots fitted with depth cameras operating at a maximum data rate of approximately 25.04 Mbps (3.13 MB/s - the average of jackal, legged and aerial robots), as shown in Table 4.

The choice of robot type can significantly impact the size of the deployable fleet. If bandwidth is a limiting factor, using robots with lower data rates, such as type A, would allow for a larger fleet. For example, given an available bandwidth of 60 Mbps, up to 16 robots of type A could be deployed, while the number of deployable robots of type B would be limited to two. On the other hand, if the vision capabilities of the robots are more important, then fewer robots of a higher data rate type, such as type B, could be deployed. To optimize the size of the fleet, a combination of both types of robots can also be considered, striking a balance between vision capabilities and fleet size.

Additionally, bandwidth management techniques can further enhance fleet optimization. A type of bandwidth management techniques, which involves controlling frequencies, enhances the system's real-time capability to adapt to changing conditions and operational priorities, while avoiding exceeding the available bandwidth. Nine steps, given in Table 6, demonstrate the coordination of bandwidth usage among robots. Steps 1–5 prioritize equal bandwidth usage across the fleet to maximize utilization when all robots are active. Steps 6–9 show the reduction of priority and frequency for each robot as it completes its mission. The frequency of the aerial robot is reduced to 1 Hz in Step 6, followed by Scout2, Legged, and Jackal in Steps 7, 8, and 9, respectively. The remaining active robots then share the allocated frequencies.

The centralized approach of the framework may result in challenges with reliability and resiliency in large areas due to interference from structures, obstacles, and wireless devices. To improve coverage and increase resiliency, wireless mesh networks [90] and decentralized communication techniques can be used, allowing for dynamic connection and disconnection capabilities. The bandwidth management algorithm allocates frequencies based on the assumption of prior knowledge of total bandwidth, but in physical settings, physical and digital twin connectivity can be affected by various factors, such as wireless connection settings and signal strength. Thus, experiments in real-world scenarios requires the approximation of prior knowledge of bandwidth.

As the results suggest, smart data management, not necessarily 5G/6G technology, is the key in overcoming bandwidth limitations and the increasing number of robots in a multi-robot fleet. To ensure seamless communication and control, the system must provide adequate bidirectional bandwidth and low latency for all active robots

Table 6
Frequencies and bandwidth usage during experimental scenario with available bandwidth of 60 Mbps (7.50 MB/s).

| Step	Scout			Jackal			Legged			Aerial		
	Freq. (Hz)	Band-width (MB/s)	Band-width (%)	Freq. (Hz)	Band-width (MB/s)	Band-width (%)	Freq. (Hz)	Band-width (MB/s)	Band-width (%)	Freq. (Hz)	Band-width (MB/s)	Band-width (%)
1	30.0	1.01	13.47	13.15	1.87	25.0	30.0	1.81	24.18	28.02	1.87	25.0
2	30.0	1.00	13.38	13.49	1.87	25.0	29.52	1.87	25.0	27.73	1.87	25.0
3	30.0	1.45	19.34	25.11	1.87	25.0	30.0	1.87	25.0	22.07	1.87	25.0
4	30.0	1.63	21.75	23.42	1.87	25.0	30.0	0.66	8.92	28.15	1.87	25.0
5	30.0	1.40	18.79	26.31	1.87	25.0	30.0	1.73	23.17	25.06	1.87	25.0
6	30.0	1.50	20.09	30.0	1.70	22.78	30.0	1.35	18.13	1.0	0.05	0.76
7	1.0	0.04	0.56	30.0	1.48	19.82	30.0	1.31	17.53	1.0	0.06	0.81
8	1.0	0.04	0.55	30.0	2.09	27.89	1.0	0.03	0.5	1.0	0.06	0.81
9	1.0	0.04	0.55	1.0	0.08	1.06	1.0	0.03	0.48	1.0	0.06	0.81
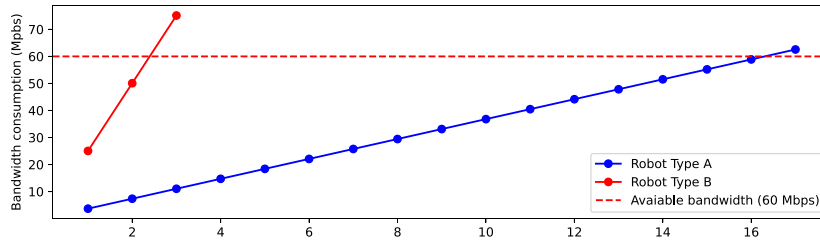
Fig. 13. Maximum number of deployable robots for both types A and B, given a 60 Mbps bandwidth and assuming uniform bandwidth distribution.

accessing the network simultaneously. This can be achieved through further elegant data usage that can adapt to changing environments and bandwidth constraints, such as trigger or change detection, priority schemes based on speed and distance to obstacles [81], and other bandwidth optimization strategies.

Moreover, low latency is crucial for multi-robot fleets to meet the requirements of interoperability, near real-time visualization, and high-rate control. The level of autonomy and latency needs can be determined based on the predicted inspection tasks and facilities. For instance, conditional autonomy (level 3) [91] may be preferred for exploring hazardous environments, as it enables the digital twin or human operator to identify multiple inspection routes.

6. Conclusion

Inspection robots play a crucial role in the operation and maintenance of asset management activities. Additionally, the development of a cyber–physical system for remote operation of these robots has the potential to provide many benefits to the field. By incorporating a fleet of heterogeneous robots with different navigation capabilities, such as wheeled, legged, and aerial, these robots can efficiently and accurately identify and perform cooperative repeated and subsequent inspection tasks in a wide range of industrial environments.

In our prior work [47], we developed a synchronous digital twin platform using a ground robot in a 3D mapping of a real-world scenario. This paper presents a cyber–physical architecture that addresses key components to ensure successful implementation and scalable governance of multi-robot fleets. This framework combines individual robots into a useful fleet with seamless communication to provide increased value to inspection missions. Eventually, this system of systems will become a valuable and integrated solution where operators can monitor and manage an autonomous fleet of robots as they perform their missions. The proposed architecture includes putting off-the-shelf techniques together in a unique way by assessing and managing bandwidth to have low-latency communication networks that are scalable to support cooperative inspection missions and synchronous bi-directional data provision across multi-robot fleets, the addition of digital twin technology, and heterogeneous multi-robot 3D mapping. The human operator can acquire common 3D map streams and visualize data of the

unknown environment through a digital twin interface with real-time communication, allowing for usable supervisory control. This framework has the potential to significantly improve overall productivity, aid in better and more efficient mission planning, enhance inspection and maintenance across various industrial applications, and support the long-term health of employees by reducing the risks of accidents, and the costs associated with them.

One limitation of the proposed cyber–physical architecture is that it may not fully address the resiliency of the communication network in the environments in which the robots are operating. This could lead to unexpected issues or challenges during implementation and use of the multi-robot fleets. To address this issue, decentralized, resilient multi-robot networks with dynamic connection and disconnection capabilities, such as those using advanced multi-master ROS networks like nimbro-networks [92] and Pound [93], can be tolerant to communication disruptions or failures. Also, packet loss prediction strategies can be used to compensate for dropped or missed packets.

Our vision for the future includes fleets of autonomous robots living within operating sites such as nuclear power plants or offshore wind farms, running long-term inspection missions. To support this robot ecosystem, robots specialized in different tasks or equipped with different sensors and tools can contribute to the sustainability effects. For example, one robot may be equipped with cameras and sensors to capture data about the environment, while another robot may be equipped with tools to perform maintenance and repairs. Future research will extend our approach with long-distance experiments, real-world testing, and industry adoption. Smart data management with AI and prediction capabilities, combined with data minimization techniques such as downsampling and compression, could further reduce bandwidth usage and enhance reliability for fleets of robots. The implementation of a Data Distribution Service (DDS) on the ROS2 platform will be explored to enhance multi-robot communication networking. Furthermore, we will focus on understanding the semantic level of asset management activities within inspection, maintenance and repair, such as identifying categories of objects and materials in scenes, semantic labeling of 3D scene models, identifying areas that need regular inspection, and verifying the state of the facility before it becomes accessible to humans.
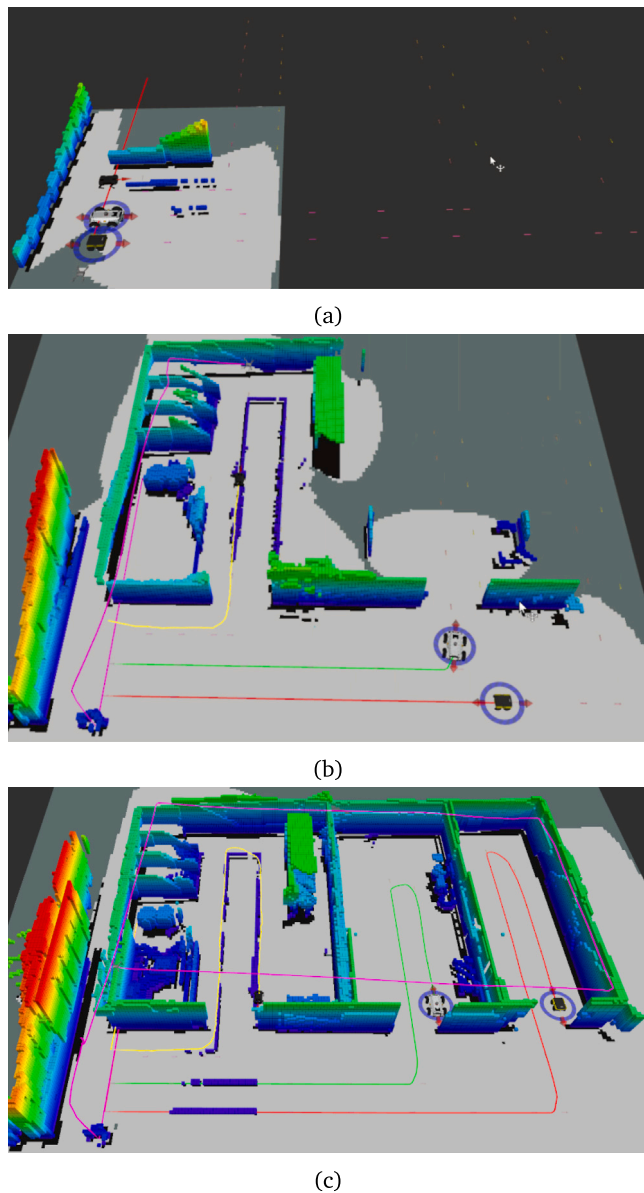
(a)

(b)

(c)

Fig. 14. A top-down view of heterogeneous multi-robot 3D voxel grid map (resolution 0.05 m). The trajectories of legged yellow, scout green, Jackal red, UAV purple.

CRediT authorship contribution statement

Hasan Kivrak: Conceptualization, Investigation, Methodology, Software, Validation, Writing – original draft. **Muhammed Zahid Karakusak:** Conceptualization, Methodology, Writing – review & editing. **Simon Watson:** Project administration, Supervision, Writing – review & editing. **Barry Lennox:** Funding acquisition, Project administration.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Data will be made available on request.

References

[1] F. Cunha, K. Youcef-Toumi, Ultra-wideband radar for robust inspection drone in underground coal mines, in: 2018 IEEE International Conference on Robotics and Automation, ICRA, IEEE, 2018, pp. 86–92.

[2] C. Papachristos, S. Khattak, F. Mascarich, K. Alexis, Autonomous navigation and mapping in underground mines using aerial robots, in: 2019 IEEE Aerospace Conference, IEEE, 2019, pp. 1–8.

[3] J.M. Aitken, M.H. Evans, R. Worley, S. Edwards, R. Zhang, T. Dodd, L. Mihaylova, S.R. Anderson, Simultaneous localization and mapping for inspection robots in water and sewer pipe networks: A review, IEEE Access 9 (2021) 140173–140198.

[4] Y. Yu, A. Safari, X. Niu, B. Drinkwater, K.V. Horoshenkov, Acoustic and ultrasonic techniques for defect detection and condition monitoring in water and sewerage pipes: A review, Appl. Acoust. 183 (2021) 108282.

[5] T.L. Nguyen, A. Blight, A. Pickering, G.H. Jackson-Mills, J.H. Boyle, R. Richardson, M. Dogar, N. Cohen, Autonomous control for miniaturized mobile robots in unknown pipe networks, Front. Robotics AI (2022) 309.

[6] A. Bouman, M.F. Ginting, N. Alatur, M. Palieri, D.D. Fan, T. Touma, T. Pailevanian, S.-K. Kim, K. Otsu, J. Burdick, et al., Autonomous spot: Long-range autonomous exploration of extreme environments with legged locomotion, in: 2020 IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS, IEEE, 2020, pp. 2518–2525.

[7] H. Azpúrua, M. Saboia, G.M. Freitas, L. Clark, A.-a. Agha-mohammadi, G. Pessin, M.F. Campos, D.G. Macharet, A survey on the autonomous exploration of confined subterranean spaces: Perspectives from real-word and industrial robotic deployments, Robot. Auton. Syst. (2022) 104304.

[8] M. Nancekievill, A. Jones, M. Joyce, B. Lennox, S. Watson, J. Katakura, K. Okumura, S. Kamada, M. Katoh, K. Nishimura, Development of a radiological characterization submersible ROV for use at Fukushima Daiichi, IEEE Trans. Nucl. Sci. 65 (9) (2018) 2565–2572.

[9] J. Lim, H. Kim, Y. Park, Review of the regulatory periodic inspection system from the viewpoint of defense-in-depth in nuclear safety, Nucl. Eng. Technol. 50 (7) (2018) 997–1005.

[10] J. Petereit, J. Beyerer, T. Asfour, S. Gentes, B. Hein, U.D. Hanebeck, F. Kirchner, R. Dillmann, H.H. Götting, M. Weiser, et al., ROBDEKON: Robotic systems for decontamination in hazardous environments, in: 2019 IEEE International Symposium on Safety, Security, and Rescue Robotics, SSRR, IEEE, 2019, pp. 249–255.

[11] P. Aivaliotis, Z. Arkouli, K. Georgoulias, S. Makris, Degradation curves integration in physics-based models: Towards the predictive maintenance of industrial robots, Robot. Comput.-Integr. Manuf. 71 (2021) 102177, http://dx.doi.org/10.1016/j.rcim.2021.102177.

[12] K. Qian, A. Song, J. Bao, H. Zhang, Small teleoperated robot for nuclear radiation and chemical leak detection, Int. J. Adv. Robot. Syst. 9 (3) (2012) 70.

[13] B. Bird, A. Griffiths, H. Martin, E. Codres, J. Jones, A. Stancu, B. Lennox, S. Watson, X. Poteau, A robot to monitor nuclear facilities: Using autonomous radiation-monitoring assistance to reduce risk and cost, IEEE Robot. Autom. Mag. 26 (1) (2018) 35–43.

[14] T. Wright, A. West, M. Licata, N. Hawes, B. Lennox, Simulating ionising radiation in gazebo for robotic nuclear inspection challenges, Robotics 10 (3) (2021) 86.

[15] K. Groves, E. Hernandez, A. West, T. Wright, B. Lennox, Robotic exploration of an unknown nuclear environment using radiation informed autonomous navigation, Robotics 10 (2) (2021) 78.

[16] I. Tsitsimpelis, C.J. Taylor, B. Lennox, M.J. Joyce, A review of ground-based robotic systems for the characterization of nuclear environments, Prog. Nucl. Energy 111 (2019) 109–124.

[17] B. Bird, M. Nancekievill, A. West, J. Hayman, C. Ballard, W. Jones, S. Ross, T. Wild, T. Scott, B. Lennox, Vega—A small, low cost, ground robot for nuclear decommissioning, J. Field Robotics 39 (3) (2022) 232–245.

[18] D. Thakur, G. Loianno, W. Liu, V. Kumar, Nuclear environments inspection with micro aerial vehicles: Algorithms and experiments, in: International Symposium on Experimental Robotics, Springer, 2020, pp. 191–200.

[19] L.R. Pinto, A. Vale, Y. Brouwer, J. Borbinha, J. Corisco, R. Ventura, A.M. Silva, A. Mourato, G. Marques, Y. Romanets, et al., Radiological scouting, monitoring and inspection using drones, Sensors 21 (9) (2021) 3143.

[20] K. Groves, A. West, K. Gornicki, S. Watson, J. Carrasco, B. Lennox, Mallard: An autonomous aquatic surface vehicle for inspection and monitoring of wet nuclear storage facilities, Robotics 8 (2) (2019) 47.

[21] M. Nancekievill, J. Espinosa, S. Watson, B. Lennox, A. Jones, M.J. Joyce, J.-i. Katakura, K. Okumura, S. Kamada, M. Katoh, et al., Detection of simulated fukushima daichii fuel debris using a remotely operated vehicle at the naraha test facility, Sensors 19 (20) (2019) 4602.

[22] A. West, I. Tsitsimpelis, M. Licata, A. Jazbec, L. Snoj, M.J. Joyce, B. Lennox, Use of Gaussian process regression for radiation mapping of a nuclear reactor with a mobile robot, Sci. Rep. 11 (1) (2021) 1–11.

[23] G. Baker, T. Bridgwater, P. Bremner, M. Giuliani, Towards an immersive user interface for waypoint navigation of a mobile robot, 2020, arXiv preprint arXiv:2003.12772.

[24] M. Ling, J. Huo, G.V. Moiseev, L. Hu, Y. Xiao, Multi-robot collaborative radioactive source search based on particle fusion and adaptive step size, Ann. Nucl. Energy 173 (2022) 109104.

[25] Y. Sahni, J. Cao, S. Jiang, Middleware for multi-robot systems, in: Mission-Oriented Sensor Networks and Systems: Art and Science: Volume 2: Advances, Springer, 2019, pp. 633–673.

[26] J.P. Queralta, J. Taipalmaa, B.C. Pullinen, V.K. Sarker, T.N. Gia, H. Tenhunen, M. Gabbouj, J. Raitoharju, T. Westerlund, Collaborative multi-robot search and rescue: Planning, coordination, perception, and active vision, IEEE Access 8 (2020) 191617–191643.

[27] M. Kulkarni, M. Dharmadhikari, M. Tranzatto, S. Zimmermann, V. Reijgwart, P. De Petris, H. Nguyen, N. Khedekar, C. Papachristos, L. Ott, et al., Autonomous teamed exploration of subterranean environments using legged and aerial robots, in: 2022 International Conference on Robotics and Automation, ICRA, IEEE, 2022, pp. 3306–3313.

[28] J.J. Roldán, P. Garcia-Aunon, M. Garzón, J. De León, J. Del Cerro, A. Barrientos, Heterogeneous multi-robot system for mapping environmental variables of greenhouses, Sensors 16 (7) (2016) 1018.

[29] M. Garzón, J. Valente, J.J. Roldán, D. Garzón-Ramos, J.d. León, A. Barrientos, J.d. Cerro, Using ros in multi-robot systems: Experiences and lessons learned from real-world field tests, in: Robot Operating System, ROS, Springer, 2017, pp. 449–483.

[30] J.K. Verma, V. Ranga, Multi-robot coordination analysis, taxonomy, challenges and future scope, J. Intell. Robot. Syst. 102 (1) (2021) 1–36.

[31] A. Kolling, P. Walker, N. Chakraborty, K. Sycara, M. Lewis, Human interaction with robot swarms: A survey, IEEE Trans. Hum.-Mach. Syst. 46 (1) (2015) 9–26.

[32] GOV.UK, Enabling a National Cyber-Physical Infrastructure to Catalyse Innovation, Tech. Rep., GOV.UK, 2022, URL: https://www.gov.uk/government/consultations/enabling-a-national-cyber-physical-infrastructure-to-catalyse-innovation. (Last Accessed 14 February 2024).

[33] A. Fuller, Z. Fan, C. Day, C. Barlow, Digital twin: Enabling technologies, challenges and open research, IEEE Access 8 (2020) 108952–108971.

[34] I. Errandonea, S. Beltrán, S. Arrizabalaga, Digital twin for maintenance: A literature review, Comput. Ind. 123 (2020) 103316.

[35] A. Rasheed, O. San, T. Kvamsdal, Digital twin: Values, challenges and enablers from a modeling perspective, IEEE Access 8 (2020) 21980–22012.

[36] P. Bremner, T.J. Mitchell, V. McIntosh, The impact of data sonification in virtual reality robot teleoperation, Front. Virtual Reality 3 (2022) 904720.

[37] P. Adjei, R. Montasari, A critical overview of digital twins, Int. J. Strategic Eng. (IJoSE) 3 (2) (2020) 51–61.

[38] D.M. Botín-Sanabria, A.-S. Mihaita, R.E. Peimbert-García, M.A. Ramírez-Moreno, R.A. Ramírez-Mendoza, J.d.J. Lozoya-Santos, Digital twin technology challenges and applications: A comprehensive review, Remote Sens. 14 (6) (2022) 1335.

[39] J. Gielis, A. Shankar, A. Prorok, A critical review of communications in multi-robot systems, Curr. Report. Rep. 3 (4) (2022) 213–225.

[40] X. Fan, C. Huang, B. Fu, S. Wen, X. Chen, et al., UAV-assisted data dissemination in delay-constrained VANETs, Mob. Inf. Syst. 2018 (2018).

[41] N. Kottege, S. Scherer, J. Faigl, A. Agha, Editorial: Special issue on advancements and lessons learned during phases I and II of the DARPA subterranean challenge, Field Robotics 2 (2022) 1947–1950.

[42] K. Ebadi, L. Bernreiter, H. Biggie, G. Catt, Y. Chang, A. Chatterjee, C.E. Denniston, S.-P. Deschênes, K. Harlow, S. Khattak, et al., Present and future of SLAM in extreme environments: The DARPA SubT challenge, IEEE Trans. Robot. (2023).

[43] B. Morrell, R. Thakker, À. Santamaria Navarro, A. Bouman, X. Lei, J. Edlund, T. Pailevanian, T.S. Vaquero, Y.L. Chang, T. Touma, et al., NeBula: TEAM CoSTAR's robotic autonomy solution that won phase II of DARPA subterranean challenge, Field Robotics 2 (2022) 1432–1506.

[44] M.F. Ginting, K. Otsu, J.A. Edlund, J. Gao, A.-A. Agha-Mohammadi, CHORD: Distributed data-sharing via hybrid ROS 1 and 2 for multi-robot exploration of large-scale complex environments, IEEE Robot. Autom. Lett. 6 (3) (2021) 5064–5071.

[45] M. Tranzatto, F. Mascarich, L. Bernreiter, C. Godinho, M. Camurri, S. Khattak, T. Dang, V. Reijgwart, J. Loeje, D. Wisth, et al., Cerberus: Autonomous legged and aerial robotic exploration in the tunnel and urban circuits of the darpa subterranean challenge, Field Robotics 2 (2022) 274–324.

[46] T. Roucek, M. Pecka, P. Cızek, T. Petrıcek, J. Bayer, V. Šalansky, T. Azayev, D. Hert, M. Petrlık, T. Báca, et al., System for multi-robotic exploration of underground environments CTU-CRAS-NORLAB in the DARPA Subterranean challenge, Field Robotics 2 (2022) 1779–1818.

[47] H. Kivrak, P.D.E. Baniqued, S. Watson, B. Lennox, An investigation of the network characteristics and requirements of 3D environmental digital twins for inspection robots, in: 2022 IEEE 23rd International Symposium on a World of Wireless, Mobile and Multimedia Networks, WoWMoM, IEEE, 2022, pp. 596–600.

[48] J.A.S. Martins, MRSLAM-Multi-Robot Simultaneous Localization and Mapping, University of Coimbra, 2013, (Master's Thesis).

[49] M.T. Lazaro, L.M. Paz, P. Pinies, J.A. Castellanos, G. Grisetti, Multi-robot SLAM using condensed measurements, in: 2013 IEEE/RSJ International Conference on Intelligent Robots and Systems, IEEE, 2013, pp. 1069–1076.

[50] Y. Yue, C. Yang, Y. Wang, J. Zhang, M. Wen, X. Tang, H. Zhang, D. Wang, Multi-robot map fusion framework using heterogeneous sensors, in: 2019 IEEE International Conference on Cybernetics and Intelligent Systems (CIS) and IEEE Conference on Robotics, Automation and Mechatronics, RAM, IEEE, 2019, pp. 536–541.

[51] M. Drwiega, 3D maps integration based on overlapping regions matching, J. Autom. Mob. Robot. Intell. Syst. 15 (3) (2022) 70–80.

[52] E. Uslu, F. Cakmak, N. Altuntaş, S. Marangoz, M.F. Amasyalı, S. Yavuz, An architecture for multi-robot localization and mapping in the gazebo/robot operating system simulation environment, Simulation 93 (9) (2017) 771–780.

[53] G.S. Martins, D. Portugal, R.P. Rocha, MRGS: A multi-robot SLAM framework for ROS with efficient information sharing, in: Robot Operating System, ROS, Springer, 2021, pp. 45–75.

[54] R. Dubé, A. Gawel, H. Sommer, J. Nieto, R. Siegwart, C. Cadena, An online multi-robot SLAM system for 3D LiDARs, in: 2017 IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS, IEEE, 2017, pp. 1004–1011.

[55] A. Hornung, K.M. Wurm, M. Bennewitz, C. Stachniss, W. Burgard, OctoMap: An efficient probabilistic 3D mapping framework based on octrees, Autonomous Robots 34 (3) (2013) 189–206.

[56] K. Yamaguchi, T. Kunii, K. Fujimura, H. Toriya, Octree-related data structures and algorithms, IEEE Comput. Graph. Appl. 4 (01) (1984) 53–59.

[57] D. Meagher, Geometric modeling using octree encoding, Comput. Graph. Image Process. 19 (2) (1982) 129–147.

[58] Webviz, 2022, https://github.com/cruise-automation/webviz. (Accessed: 14 February 2024).

[59] ROSboard, 2022, https://github.com/dheera/rosboard. (Accessed: 14 February 2024).

[60] Rosshow, 2022, https://github.com/dheera/rosshow. (Accessed: 14 February 2024).

[61] Foxglove, 2022, https://foxglove.dev/. (Accessed: 14 February 2024).

[62] Formant, 2022, https://formant.io/. (Accessed: 14 February 2024).

[63] Rocos-drone deploy, 2022, https://www.dronedeploy.com/. (Accessed: 14 February 2024).

[64] Freedom robotics, 2022, https://www.freedomrobotics.com/. (Accessed: 30 November 2022).

[65] Rqt, 2022, https://github.com/ros-visualization/rqt. (Accessed: 14 February 2024).

[66] Rosbridge suite, 2019, http://wiki.ros.org/rosbridge_suite. (Accessed: 14 February 2024).

[67] Rqt image view, 2020, http://wiki.ros.org/rqt_image_view. (Accessed: 14 February 2024).

[68] Rviz, 2019, http://wiki.ros.org/rqt_rviz. (Accessed: 14 February 2024).

[69] Rqt multi plot, 2016, http://wiki.ros.org/rqt_multiplot. (Accessed: 14 February 2024).

[70] The cyber-physical fabric summit, 2021, https://www.cdbb.cam.ac.uk/news/cyber-physical-fabric-summit, Online conference, Royal Academy of Engineering, BEIS, GoScience, UKRI, Alan Turing Institute.

[71] IEEE 802.11 Wi-Fi generations, 2023, https://en.wikipedia.org/wiki/IEEE_802.11. (Accessed: 30 January 2023).

[72] M. Giordani, M. Polese, M. Mezzavilla, S. Rangan, M. Zorzi, Toward 6G networks: Use cases and technologies, IEEE Commun. Mag. 58 (3) (2020) 55–61.

[73] C.-H. Hsu, U. Kremer, IPERF: A framework for automatic construction of performance prediction models, in: Workshop on Profile and Feedback-Directed Compilation, PFDC, Paris, France, Citeseer, 1998.

[74] J. Kammerl, N. Blodow, R.B. Rusu, S. Gedikli, M. Beetz, E. Steinbach, Real-time compression of point cloud streams, in: 2012 IEEE International Conference on Robotics and Automation, 2012, pp. 778–785, http://dx.doi.org/10.1109/ICRA.2012.6224647.

[75] R.B. Rusu, S. Cousins, 3D is here: Point cloud library (pcl), in: 2011 IEEE International Conference on Robotics and Automation, IEEE, 2011, pp. 1–4.

[76] F. Pomerleau, F. Colas, R. Siegwart, S. Magnenat, Comparing ICP variants on real-world data sets, Auton. Robots 34 (3) (2013) 133–148.

[77] J. Dybedal, A. Aalerud, G. Hovland, Embedded processing and compression of 3d sensor data for large scale industrial environments, Sensors 19 (3) (2019) 636.

[78] T. Wiemann, F. Igelbrink, S. Pütz, M.K. Piening, S. Schupp, S. Hinderink, J. Vana, J. Hertzberg, Compressing ROS sensor and geometry messages with draco, in: 2019 IEEE International Symposium on Safety, Security, and Rescue Robotics, SSRR, IEEE, 2019, pp. 243–248.

[79] S. Pacheco-Gutierrez, H. Niu, I. Caliskanelli, R. Skilton, A multiple level-of-detail 3D data transmission approach for low-latency remote visualisation in teleoperation tasks, Robotics 10 (3) (2021) 89.

[80] S. Pacheco-Gutierrez, I. Caliskanelli, R. Skilton, Point cloud compression and transmission for remote handling applications, J. Softw 16 (2021) 14–23.

[81] R.E. Julio, G.S. Bastos, Dynamic bandwidth management library for multi-robot systems, in: 2015 IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS, IEEE, 2015, pp. 2585–2590.

[82] Nuclear Decommissioning Authority's grand challenges, 2022, https://www.gov.uk/government/news/nda-sets-out-its-grand-challenges. (Accessed: 14 February 2024).

[83] Champ, 2022, https://github.com/chvmp/champ. (Accessed: 14 February 2024).

[84] J. Lee, et al., Hierarchical Controller for Highly Dynamic Locomotion Utilizing Pattern Modulation and Impedance Control: Implementation on the MIT Cheetah robot (Ph.D. thesis), Massachusetts Institute of Technology, 2013.

[85] L. Meier, P. Tanskanen, F. Fraundorfer, M. Pollefeys, Pixhawk: A system for autonomous flight using onboard computer vision, in: 2011 IEEE International Conference on Robotics and Automation, IEEE, 2011, pp. 2992–2997.

[86] R. Prasad, C. Dovrolis, M. Murray, K. Claffy, Bandwidth estimation: Metrics, measurement techniques, and tools, IEEE Netw. 17 (6) (2003) 27–35.

[87] Ros topic, 2012, http://wiki.ros.org/rostopic. (Accessed: 14 February 2024).

[88] C. Bovy, H. Mertodimedjo, G. Hooghiemstra, H. Uijterwaal, P. Van Mieghem, Analysis of end-to-end delay measurements in internet, in: Proc. of the Passive and Active Measurement Workshop-PAM, vol. 2002, sn, 2002.

[89] P. Pandey, R. Parasuraman, Empirical analysis of bi-directional wi-fi network performance on mobile robots in indoor environments, in: 2022 IEEE 95th Vehicular Technology Conference, VTC2022-Spring, IEEE, 2022, pp. 1–7.

[90] Rajant, The Power of Rajant InstaMesh with LTE/5G, Tech. Rep., 2022, URL: https://airadio.com/download?file=mediaPool/uAl13zL.pdf. (Last Accessed 14 February 2024).

[91] S. Watson, D.A. Duecker, K. Groves, Localisation of unmanned underwater vehicles (UUVs) in complex and confined environments: A review, Sensors 20 (21) (2020) 6203.

[92] GitHub - Nimbro_network: ROS transport for high-latency, low-quality networks, 2023, https://github.com/AIS-Bonn/nimbro_network. (Accessed: 14 February 2024).

[93] D. Tardioli, R. Parasuraman, P. Ögren, Pound: A multi-master ROS node for reducing delay and jitter in wireless multi-robot networks, Robot. Auton. Syst. 111 (2019) 73–87.