A Telexistence Interface for Remote Control of a Physical Industrial Robot via Data Distribution Service

Damien Mazeas[1](✉), John Ahmet Erkoyuncu[1], and Frédéric Noël[2]

[1] Centre for Digital Engineering and Manufacturing, Cranfield University, Cranfield, UK
Damien.j.mazeas@cranfield.ac.uk
[2] G-SCOP, Grenoble University, Grenoble, France

Abstract. The increasing adoption of remote working practices and internet-based systems highlights the need for a new mode of interaction and communication interfaces between humans and machines. Current human-machine interfaces (HMI) are based on complex controllers that are not intuitive for a novice or remote operators. The increase in skills required in today's manufacturing environment allows the use of telexistence capabilities to facilitate human-robot collaboration. We propose a digital-twin based framework with a telexistence interface as a means to reduce the complexity of operating and programming an industrial robot. To remotely control the industrial robot, sensors data are shared via a data distribution service (DDS). The immersive virtual reality (VR) interface is deployed for effective control and monitoring of the industrial robot. Telexistence capabilities allow intuitive manipulation and are combined with real-time data visualization of the robot through the digital twin. The interface is implemented with the Unity 3D engine and connected to a console application that collects sensor data and shares them via a DDS connectivity framework. A simple experiment with a physical FANUC M20-IA industrial robot and Azure Kinect RGBD cameras shows the reliable performance when a robot path request was sent by a remote operator through DDS and the immersive VR interface. We then discuss future work and use cases for this telexistence platform to support maintenance activities in manufacturing contexts.

Keywords: Telexistence · Virtual reality · Digital twin · Immersive interface

1 Introduction

Working environments can be potentially hazardous to humans in many sectors such as nuclear engineering, manufacturing, or defence. Operators are exposed to heavy equipment, fast-moving machinery, or harmful materials. To maintain a safe working environment and provide flexible solutions while improving efficiency throughout the product life cycle, companies are responding with a smart factory vision. Although the main objective of robots in automation is to provide the highest level of autonomy to perform complex repetitive tasks such as disassembly or sorting, when it comes to unplanned

Published by Springer Nature Switzerland AG 2023
F. Noël et al. (Eds.): PLM 2022, IFIP AICT 667, pp. 388–398, 2023.
https://doi.org/10.1007/978-3-031-25182-5_38

situations, automation is limited. Providing industrial robots with the ability to deal with unexpected events is essential for the effectiveness and productivity of the factory. Humans can exploit experiences in their fields to provide innovative solutions and use maintenance robots to do so. To avoid wasting time and money, it is important to keep the Human in the loop by enabling safe remote collaboration between the human and the robot. We hypothesize that the introduction of telexistence capability into remote work on tasks such as maintenance can enhance the capabilities and way of working for remote maintainers since it could lead to a more intuitive user interface (UI) and better spatial awareness, due to advantages such as sense of presence and more advanced visualization of the remote asset to maintain with the use of the immersive digital twin. A prototype that reflects the real-time operating conditions of a physical asset is a twin [1], Grieves [2] has described the concept of digital twin as a real and virtual spaces with a link for data flow from the real to the virtual space and a link for information flow from the virtual space to the real space. Lack of appropriate visualization of interaction forms is one of the main gaps in digital twins' literature reviews and immersive technologies are often cited to overcome these issues [3]. Conventional methods such as video feed-based teleoperation do not provide sufficient environment awareness and an efficient UI for the various data involved. A 2D control screen can be confusing for an operator of a remote robot [4]. Extended reality (XR), which refers to immersive 3D technologies such as virtual reality (VR) has raised the interest to support remote maintenance workers [5, 6]. Human-robot interaction in the work environment is a widely researched topic in robotics. XR has been identified as a useful tool to provide the operator with immersive real-time feedback to facilitate remote robot control [7–9]. Combining haptic devices and sensors allows for better operability of the system and a sense of presence for the operator in the simulated remote environment, whilst enhancing telexistence capability. This concept was proposed by Professor Susumu Tachi in 1980 [10], it is described as a fusion of VR and robotics which enables a sense of existence in another place.

Therefore, the overall goal of the present study is to develop a reliable framework via data distribution service (DDS) and implement telexistence capability for an industrial robot FANUC. Industrial robots are fully automated, pre-programmed for their tasks without human interaction, while collaborative robots (cobots) work with humans. As part of the industry 4.0, we want to bring human and industrial robot in a same collaborative and single workplace thanks to a telexistence interface. Our framework aims to offer usability and functionality to make it suitable specifically for operating and monitoring a physical industrial robot through its immersive digital twin in real-time. Through the design of this framework, we want to investigate whether telexistence-based remote monitoring and control interface can be applied to an industrial robotic arm using a modern data centric protocol and the immersive visualization of its digital twin.

2 Telexistence DDS-Based Framework

2.1 Interface Architecture

The following interface architecture has been developed to remotely enable the communication between the FANUC industrial robot and the human operator as well as to visualize the spatial context. This spatial context includes the static CAD model of the robot equipped with a gripper, dynamic data from the robot actuators and Kinect video/3D sensors and the operator orientation and position through the use of a VR Head Mounted Display (HMD).

In the beginning, a search and comparison was conducted on communication standards/protocols widely used in the industry. Thus, OPC unified architecture (OPC UA), message queuing telemetry transport (MQTT), advanced message queuing protocol (AMQP) and data distribution service (DDS) were identified as relevant for our framework among others. These protocols are used to design industrial Internet of Things (IoT) applications to efficiently share data. The majority of these protocols are designed for simplicity and can support a very limited set of use cases except for DDS. Our choice to use DDS is justified by its wide field of application in real-time and scalable systems. The protocol DDS provide features in term of data-centric approach, predictability, real-time quality of service settings and respect to security [11]. The ROS2 robotic interface is also based on top of DDS. In addition, it facilitates development efforts and allows data to be efficiently delivered to systems using a publish-subscribe pattern and is an Object Management Group (OMG) machine-to-machine standard.

Figure 1 shows the architecture of our DDS-based framework. To send and receive data and commands, a publisher node is creating a topic and a subscriber node is receiving data from a specific topic. Moreover, DDS integrate a concept of Quality of Service (QoS) to configure the parameters of participants inside the system.

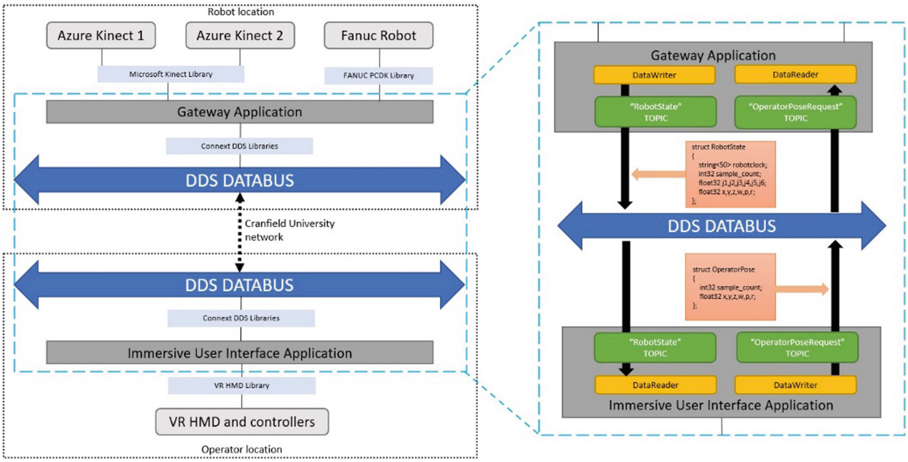


Fig. 1. Architecture and components overview of the telexistence rig framework

Subsequently, DDS was set up by creating data types sent in samples. A DDS sample is a combination of a Topic (distinguished by a Topic name) and the actual data type defined by the user. The ROS libraries, the most used operating system for robotics provides common interface messages organised in modules such as sensor devices or geometric primitives. It also offers the possibility to create custom messages. For our framework, since ROS is not implemented and for simplifying the development, we have created our own data structures. However, it could make sense in the future to match

ROS data types for interoperating with ROS applications. We created ours to provide a means to manage data efficiently for our framework especially for large amount of data such as the point cloud. Only the necessary data is transmitted to achieve the desired function and reusable through different functions (Table 1).

Table 1. Data structures description of samples transmitted

	Data structure name	Data structure members
From the FANUC robot to remote operator	Robot_State	- Clock (string - robot internal timestamp) - Sample (int - samples count) - J1, J2, J3, J4, J5, J6 (floats - joints angles in degree) - X, Y, Z (floats - robot world position) - W, P, R (floats - robot world rotation)
	Robot_Point_Cloud	- Clock (string - robot internal timestamp) - Sample (int - samples count) - Sequence Memory (floats - sequence containing points position and Colour data)
	Robot_Image	- Clock (string - robot internal timestamp) - Sample (int - samples count) Memory (bytes - sequence containing MPEG image data)
	Robot_Alarm	- Clock (string - robot internal timestamp) - Sample (int - samples count) - Message (string of alarm message)
	Robot_Reachability_State	- Clock (string - robot internal timestamp) - Sample (int - samples count) - IsReachable (bool - state regarding reachability for a position by the robot)

(*continued*)

Table 1. (*continued*)

	Data structure name	Data structure members
From the remote operator to the FANUC robot	Operator_Request	- Clock (string - operator computer timestamp) - Sample (int - samples count) - Buttons (enumeration of buttons, RESET, ABORT, HOME, PATH)
	Operator_Teleop_Target	- Clock (string - operator computer timestamp) - Sample (int - samples count) - X, Y, Z (floats - operator target position) - W, P, R (floats - operator target rotation)
	Operator_Path_Point	- Clock (string - operator computer timestamp) - Sample (int - samples count) - ID (int - added/modified pose ID) - IsUpdating (boolean - sample is for updating point values) - IsDelete (boolean - sample is for deleting pose) - X, Y, Z (floats – pose position values) - W, P, R (floats - pose rotation values)

2.2 Immersive Operator Station

Operating an industrial robotic arm is not possible without experience, keeping that in mind we started developing our telexistence rig keeping functions as simple as possible for an untrained user. For this purpose, the following functionality are expected:

- A functionality to interact with the immersive UI.
- A functionality to interact with the robot digital twin.
- A functionality to jog[1] the robot.
- A functionality to register a user pose.
- A functionality to switch between robots' programs.
- A functionality to abort the current robot motion.

[1] Jogging the industrial robot is the term used to describe the act of manually moving the robot via the user interface.

The interaction in the 3D environment is done as shown in Fig. 2, a controller button is dedicated to the teleport function (left) as well as another one for a pointer (middle) and one for direct manipulation of the 3D cursor (right). The main way of programming a robot without an offline simulation software is to jog the robot using the teach pendant device, the 3D cursor will aim to replace the jogging keys on the regular FANUC HMI. The emergency stop is a controller button which is faster to trigger than the "ABORT" 3D button in the environment UI in case the user wants to stop the robot motion.

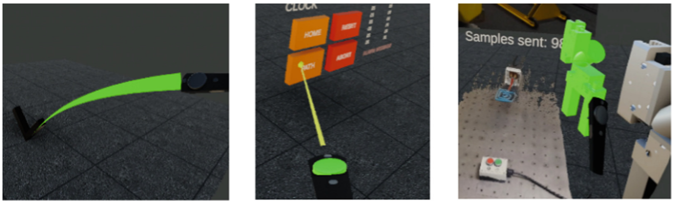


Fig. 2. Interaction methods of the framework

Figure 3 shows the elements of the UI, each element can be described as follows:

1. Four 3D buttons for the following features and the robot clock:
   - The possibility to call the home program thanks to the "HOME" button to bring the robot into its default pose.
   - The possibility to teleoperate the robot and create a path thanks to the "PATH" button. The teleoperation of the robot is done by moving the 3D cursor as seen in Fig. 2 (right). When the 3D cursor is green the robot can reach the new pose, when the 3D cursor is red the robot cannot reach the new pose and the pose is not updated in the robot program to prevent it to go into fault mode. This same "PATH" button also allows sending poses that will be used to create a path program to repeat a specific trajectory chosen by the user. This program is created on the robot gateway application by creating a Karel[2] file with a pose sent by the user and registered in the robot controller. This Karel file is then sent to the controller which compiles and executes it when the user wants to play the trajectory created.
   - The "RESET" button to reset the current robot controller fault.
   - The "ABORT" button to abort the current controller robot task.
2. A 2D video feed viewer of the first Kinect with the remote robot current ping in milliseconds. On the bottom right of the viewer, there is the number of samples received from the robot.
3. A 3D visualization of the 3D data received from the second Kinect. To do so, we have used additional Unity packages, High-Definition renders pipeline and Visual Effect (VFX) Graph, a node-based visual logic graph creator. The VFX graph created for this project takes a mesh generated from the received Kinect data by a unity C# script

[2] Lower-level language similar to Pascal and used to write FANUC robots programs.

as an input, using vertices of this mesh to output particles in cube shape. Position and Color are updated 26 times per second. The Kinect is currently calibrated manually in the 3D environment.

4. Angles value of the 6 joints of the robot in degree, world pose in millimeters and degree of the robot as well as the last alarm message are also displayed to the user.

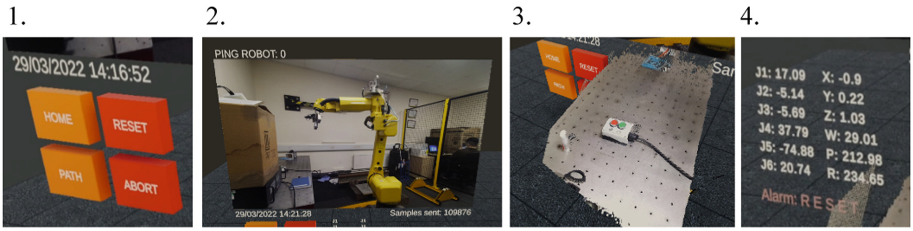


Fig. 3. UI of the framework, 3D buttons (1), 2D video feed viewer (2), point cloud (3) and robot pose values (4)

2.3 Coordinate Systems

The industrial FANUC robot's pose consists of a position XYZ in millimeters and three angles WPR in degrees with W rotation around the x-axis, P rotation around the y-axis and R rotation around the z-axis. The coordinate systems of Unity and FANUC as shown in Fig. 4 are different. The transformation from FANUC XYZ WPR to Unity 3D can be done by converting angles in degrees and then to a quaternion. We also have inverted x-axis and P, R angles to match coordinate systems.

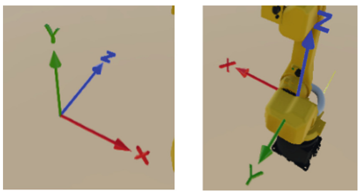
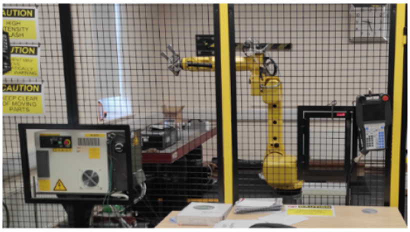


Fig. 4. Coordinate systems of Unity 3D (left) and Fanuc (right)

3 Case Study

To illustrate the proposed framework, we used a FANUC M-20iA (Fig. 5) of the Centre for Digital Engineering and Manufacturing (CDEM) at Cranfield University. It is controlled by an R-30iB mate cabinet and equipped with a pneumatic gripper. The objective



Specification		Motion Range	
Axes	6	J1	±370°
Payload	20 kg	J2	±260°
H-Reach	1811 mm	J3	±458°
Repeatability	±0.08 mm	J4	±400°
Mass	250 kg	J5	±380°
Structure	Articulated	J6	±900°
Mounting	Floor		

Fig. 5. Cranfield University FANUC robot setup (left) and the robot specifications (right)

is to test the DDS protocol over the university networks. Safety limits have been set to prevent the robot from colliding with the environment.

In the first instance, the robot is connected via Ethernet to a computer running the gateway application (Fig. 1), a static local IP address has been defined on the robot and on the computer to allow a ping command between the two. The computer is also connected to the university networks using Wi-Fi.

In a second step, at a remote place of Cranfield University a computer connected also to the university network is running the immersive UI application (Fig. 1) and is equipped with a VR HMD Varjo XR-3.

The two computers remotely located but on the same network have an XML file that describes the QoS profile of our DDS system including the 'initial peers list' (hostname of the two machines) as part of the participant phase of the discovery process. Other properties of the QoS profile are set, for instance the name participants ("Operator" and "Fanuc") and the buffers size limits. Our QoS profile is based on BEST_EFFORT_RELIABILITY_QOS which is good for periodic updates of sensor values, fast and efficiently with the least resource-intensive method. It gets the newest value for a topic from DataWriters to DataReaders (Fig. 1) without taking care of previous samples sent but it is not guaranteed that the data sent will be received. It may be lost due to connectivity issues or loss by the physical transport layer.

Finally, two Azure Kinect cameras are positioned around the robot, one targeting the table in front of the robot for the point cloud and the other targeting the robot itself for a wider view of the operating area, the two-camera run different configurations and are connected to the computer on the robot side.

The overall UI application can be started on any computer equipped with a VR HMD. Regarding the deployment of our framework on different robots, further work will be need. It is possible to keep the framework as it is for all FANUC robots only by importing and setting up the new robot CAD files in unity 3D. However, for different branded robot, the gateway application must be re-developed with the relevant libraries. Also, FANUC SDK is using a specific robot server program which must be installed to access robot controller and enable communication with the robot using TCP/IP and to the gateway application through an object (COM) interface.

4 Discussion

We developed a framework that offers simple teleoperation and trajectory creation of a FANUC industrial robot using telexistence capability and DDS-based. We focused on the interface using DDS which is an industrial protocol standard for scalable and real-time distributed systems. Our contributions include the development of the telexistence framework composed on the robot side a gateway application in C#.NET and on the remote operator side an immersive UI application in C# developed on Unity 3D game engine. This telexistence provide a safe immersive working environment because the robot and the human operator are interacting in a virtual environment, avoiding potentially harmful operations. Furthermore, the complexity of operating the robot in immersive environment can be evaluate and compare with traditional HMI or other 2D monitor based interface with our framework. Manipulating the 3D cursor in the immersive environment required less training and time than using the regular Fanuc HMI. This could be particularly advantageous when a maintenance operation must be carried out under tight delay. A further contribution is the development of the immersive digital twin for the robot set up in Unity 3D and in real-time with DDS. This is usually achieved through offline simulation software like ROBOGUIDE[3]. Lastly, we proposed a framework without the use of a Robot Operating System (ROS) which is most of the time used in a project involving robotics, especially for research, however, ROS2 has introduced the DDS protocol. We employed RTI Connext [12] which is a DDS distributor because they provide academics free of cost license and a C# SDK that is perfect for integration with Unity 3D as well as use with FANUC SDK.

When developing an immersive digital twin interface, difficulties remain with interoperability of commercial and open-source tools. The development is not straight forward, the FANUC SDK is for instance operating system dependent. The company does not provide official ROS packages; however unofficial packages exist using Karel for communication [13]. Garg et al. [14] have also recently developed a digital twin application of a FANUC robot using Karel for communication. Most of the research involving teleoperation of industrial robotics like FANUC is limited to simulation, we have implemented our framework in a real physical robot. Videos demonstrations are available[4].

Our DDS-based telexistence framework will be easy to update for other experiments, as all selected data structures are dynamically created at runtime in our applications have been created by ourselves. It has also shown reliable performance during our test using the university Networks. Further tests should be carried out on a wider network using a VPN in the future.

Limitations can also be highlighted already regarding the lack of accuracy in the VR environment. Compared to traditional interface in which user can enter precisely the value of the 3D position, immersive environment does not offer as high level of accuracy which could potentially influence operator ability to reach a certain position. To overcome these issues, we can hypothesize that different perspectives combine with the use of 3D data can improves the accuracy for complex remote tasks. Additional evaluations with participants including more complex tasks are required.

As a next step, the functionality of our framework will be enhanced and applied to an inspection maintenance scenario, developed in association with Dstl and with utility for deployed maintenance and repair operations. The 3D UI will be adapted to this use

[3] Offline motion and command simulation for FANUC branded robots.

[4] https://www.youtube.com/channel/UCyuSQ1JzesH9KpYrImPu4Fw.

case. Task performance assessment of telexistence for remote maintenance will also be performed.

5 Conclusion

The integration of telexistence capability for activities such as maintenance has the potential to offer a range of advantages such as enhanced visualizations, an enhanced sense of presence for a user operating in a remote location, as well as an intuitive interaction. It speeds up the handling of the robot without the need for extensive training with the regular FANUC HMI, we tested simple trajectory creation. The present developed framework shows the possibility of using a physical industrial robot with telexistence thanks to its immersive digital twin. The result is a hybrid immersive framework concept that allows an operator to interact with a robot normally surrounded by a fence and to program it with direct manipulation as if he were inside the fence near the robot, and this, in complete security and at a distance. Our framework is based on the Unity 3D game engine, a tool that is being more and more used in the industrial environment and for creating interactive 3D environments. Further research will be conducted to apply this framework in maintenance context tasks with user evaluation of the proposed UI, including time to complete the task, a functionality questionnaire, and a usability test.

Acknowledgment. This research is funded by the joint Anglo-French PhD scheme, organized by the respective nations' Ministry of Defence, with Dstl acting as the Technical Partner.

References

1. Rosen, R., von Wichert, G., Lo, G., Bettenhausen, K.D.: About the importance of autonomy and digital twins for the future of manufacturing. IFAC-PapersOnLine **48**, 567–572 (2015). https://doi.org/10.1016/j.ifacol.2015.06.141
2. Grieves, M., Vickers, J.: Digital twin: mitigating unpredictable, undesirable emergent behavior in complex systems. In: Kahlen, F.-J., Flumerfelt, S., Alves, A. (eds.) Transdisciplinary Perspectives on Complex Systems, pp. 85–113. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-38756-7_4
3. El Saddik, A.: Digital twins: the convergence of multimedia technologies. IEEE Multimed. **25**(2), 87–92 (2018). https://doi.org/10.1109/MMUL.2018.023121167
4. Fung, W.K., Lo, W.T., Liu, Y.H., Xi, N.: A case study of 3D stereoscopic vs. 2D monoscopic tele-reality in real-time dexterous teleoperation. In: 2005 IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS, pp. 181–186 (2005). https://doi.org/10.1109/IROS.2005.1545299
5. Linn, C., Bender, S., Prosser, J., Schmitt, K., Werth, D.: Virtual remote inspection - a new concept for virtual reality enhanced real-time maintenance. In: Proceedings of the 2017 23rd International Conference on Virtual Systems and Multimedia, VSMM, pp. 1–6 (2018). https://doi.org/10.1109/VSMM.2017.8346304
6. Masoni, R., et al.: Supporting remote maintenance in industry 4.0 through augmented reality. Procedia Manuf. **11**, 1296–1302 (2017). https://doi.org/10.1016/J.PROMFG.2017.07.257
7. Burdea, G.C.: Invited review: the synergy between virtual reality and robotics. IEEE Trans. Robot. Autom. **15**, 400–410 (1999). https://doi.org/10.1109/70.768174

8. Stotko, P., et al.: A VR system for immersive teleoperation and live exploration with a mobile robot. In: IEEE International Conference on Intelligent Robots and Systems, pp. 3630–3637 (2019).https://doi.org/10.1109/IROS40897.2019.8968598
9. Chen, J., Glover, M., Li, C., Yang, C.: Development of a user experience enhanced teleoperation approach. In: ICARM 2016 International Conference on Advanced Robotics and Mechatronics, pp. 171–177 (2016). https://doi.org/10.1109/ICARM.2016.7606914
10. Tachi, S.: Telexistence. In: Brunnett, G., Coquillart, S., van Liere, R., Welch, G., Váša, L. (eds.) Virtual Realities. LNCS, vol. 8844, pp. 229–259. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-17043-5_13
11. Talaminos-Barroso, A., Estudillo-Valderrama, M.A., Roa, L.M., Reina-Tosina, J., Ortega-Ruiz, F.: A machine-to-machine protocol benchmark for eHealth applications – use case: respiratory rehabilitation. Comput. Methods Programs Biomed. **129**, 1–11 (2016). https://doi.org/10.1016/J.CMPB.2016.03.004
12. Connext Product Suite for Intelligent Distributed Systems. https://www.rti.com/products
13. Fanuc - ROS Wiki. http://wiki.ros.org/fanuc
14. Garg, G., Kuts, V., Anbarjafari, G.: Digital twin for Fanuc robots: industrial robot programming and simulation using virtual reality. Sustainability **13**, 10336 (2021). https://doi.org/10.3390/SU131810336