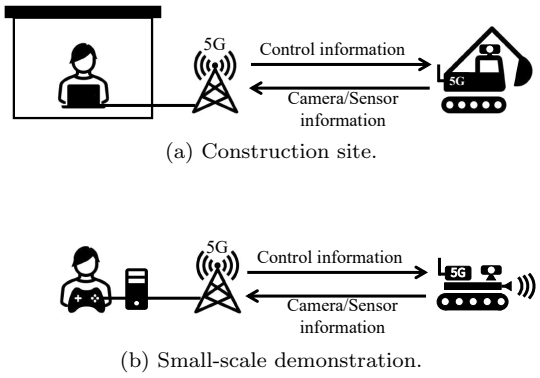
Small-scale Demonstration of Remote Control Employing Local 5G System

Issei MAKINO[†], **Junji TERAI**[††], ***Nonmembers, and*** **Nobuhiko MIKI**[†a)], ***Member***

SUMMARY Local (private) 5G system can offer the secure and flexible network by the cellular-based technologies at their facility (e.g., factories, agricultural lands, and buildings). One of the applications is remote control of the devices such as heavy machinery from the distant locations. We built up the small-scale system which demonstrates the remote control of a unmanned ground vehicle (UGV) using local 5G system. The application-level one way delay over the small-scale system using a webcam is measured to be approximately 170 ms. Furthermore, in order to estimate the wireless delay, the `ping` round trip time is also measured to be approximately 8.5 ms, which is much smaller than the webcam application delay. Finally, we used the small-scale demonstration system at the campus tour for high school students.

key words: *Local 5G, Remote control*



(a) Construction site.

(b) Small-scale demonstration.

Fig. 1: Demonstration of remote control.

1. Introduction

5G commercial services have been launched around the world, and 2.8 billion 5G subscriptions are forecast [1]. In addition to the public 5G networks, which telecommunications carriers operate, the local (private) 5G networks are now getting a lot of attention. Compared to the Wi-Fi network, which is the now natural option for the private network, the local 5G networks can offer the secure and flexible network by the cellular-based technologies [2]. It can guarantee wider coverage at their facility (factories, agricultural lands, and buildings) than Wi-Fi network. Therefore, one of the promising options for manufacturing industry. Several different application areas [3] can be distinguished:

- Factory automation
- Process automation
- Human-machine interfaces (HMIs) and production IT
- Logistics and warehousing
- Monitoring and maintenance

For each of these application areas, a multitude of potential use cases exists. One example is "mobile robots", which can be automatically or remotely controlled. In this paper, we focus on the remote control. In Japan, the operation of heavy machinery at a construction site has been shown to be a promising scenario for remote control via 5G. A large-scale field trials employing the heavy machinery at a construction site were performed. Such field trials were conducted as part of spreading the development and testing of 5G use cases in Japan across various regions, and promoting the use of 5G to find solutions to various societal issues [4].

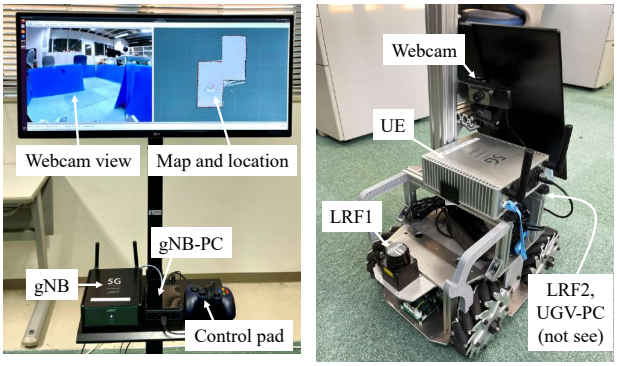
In addition to such motivation, the purpose of our demonstration is to arouse the interest of high school students and other people who will be responsible for realizing the beyond 6G in the future. We build the small-scale demonstration of remote control of an unmanned ground vehicle (UGV) employing local 5G system. Furthermore, we perform demonstration at the campus tour for high school students.

2. Demonstration Setup

2.1 Downscaling Demonstration

We first describe how to downscale the demonstration. Figure 1 schematically illustrates the remote control of construction machinery at the location near the construction site. As shown in Fig. 1(a), the construction worker remotely controls the construction machinery using a steering wheel and levers in a remote cockpit in a building near the construction site. The construction machinery is equipped with cameras and sensors, and the information collected from these is transmitted to the base station, gNB, via local 5G uplink. The views from the cockpit are reproduced, and additional information is provided. On the contrary, the control

[†]The author is with the Kagawa University, Hayashicho 2217-20, Takamatsu, Kagawa, 761-0396 Japan.
[††]The author is with STNet, Incorporated, Kasugacho 1735-3, Takamatsu, Kagawa, 761-0195 Japan.
a) E-mail: miki.nobuhiko@kagawa-u.ac.jp



(a) gNB side. (b) UE side.

Fig. 2: Demonstration equipments.

Table 1: UGV (Mecanumrover ver 2.1 [7]).

Name	Values
ROS distribution	Melodic
LRF	HOKUYO URG-04LX-UG01 [5] (240 deg.)
Number of LRF	2 (front and rear)
Webcam	Sanwa Supply CMS-V53BK (150 deg., 2 million px)

information obtained from the steering wheel and levers is sent to the construction machinery via the local 5G downlink. In our small-scale demonstration, we downscale them as shown in Fig. 1(b). The UGV is equipped with a webcam and sensors, and the information collected from these is transmitted to the base station via local 5G uplink. The video from the webcam, and the two-dimensional map and location information are provided by the PC connected to the gNB. On the contrary, the control information obtained from the control pad is sent to the UGV via the local 5G downlink.

2.2 Small-scale Demonstration Setup

Next, we will explain our small-scale demonstration setup. Figure 2 shows the demonstration equipments. As shown in Fig. 2(a), the gNB is connected to the small PC (gNB-PC). The gNB-PC is connected to the monitor and the control pad. The UE is connected to another small PC. The small PC equipped with the webcam (we call UGV-PC hereafter) is connected to the UGV. The UGV is equipped with two laser range finder (LRF) [5], one at the front and one at the rear. The robot operating system (ROS) [6] Melodic is installed on Ubuntu 18.04 as a middleware.

Figure 3 schematically illustrates the uplink and the downlink procedure of the demonstration. We describe the uplink and downlink procedure, and the local 5G system in the subsequent subsection.

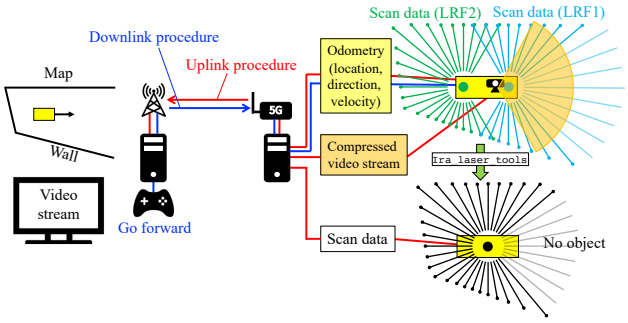


Fig. 3: Uplink and downlink procedure.

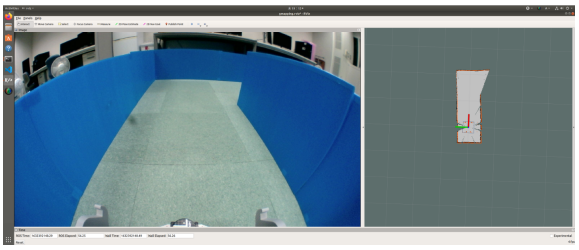
2.2.1 Uplink Procedure

The UGV-PC obtains the odometry information (location, direction, and velocity) from the control board connected to the motor of the UGV. The UGV-PC also obtains the scan information (direction, and distance) of two LRFs. Although one LRF can measure the distance to the objects with the scanning angle of ±120 degrees [5], the `ira_laser_tools` package [8] merges the two LRF scan results in a single scan result with the scanning angle of 360 degrees as shown in Fig. 3. The video stream of webcam with 1920×1080 resolution at 30 fps is generated by the `libuvc_camera` package, and compressed into jpeg format. The odometry, scan, and compressed video stream are sent to the gNB via the local 5G uplink.

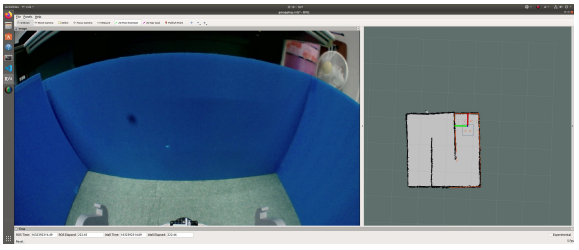
Figure 4 shows the snapshot of the gNB-PC monitor. The left side of the figure shows the video stream of the webcam sent from the UGV. Furthermore, the right side of the figure shows a map, and location and direction of the UGV, which is calculated by the `slam_gmapping` package, and visualized by the `rviz` package. The `slam_gmapping` package performs the simultaneous localization and mapping (SLAM) algorithm [9], [10] from the odometry and scan information from the UGV, and generate a 2D occupancy grid map. In the 2D grid map, the black, light gray, and dark gray grids represent the occupied (e.g. wall), free (where the UGV can move), and unknown grids, respectively. Furthermore, the orange points represent the obstacles obtained from the scan information. The red line shows the direction of the UGV. As shown in Fig. 4(a), only the area near the UGV where scan data is available is in either the free or occupied state in the initial state. By maneuvering the UGV to the other end of the maze, as shown in Fig. 4(b), the overall structure of the maze is made clear.

2.2.2 Downlink Procedure

The operator sees the video stream and the map and UGV location, and can control the UGV by the control pad to generate one of the following six commands:



(a) Initial state.



(b) Final state.

Fig. 4: gNB-PC monitor snapshot.

Table 2: Experimental Setup of local 5G system.

Name	Values
Carrier frequency	4.8499 GHz
Bandwidth	100 MHz (264 RBs)
Maximum transmission power	23 dBm
Duplex	TDD
Subcarrier spacing	30 kHz

go forward/backward/left/right, *turn left/right*. The *go* and *turn* command set the constant speed and the constant rotational speed in the odometry. The gNB-PC sends this odometry to the UGV-PC via the local 5G downlink. After receiving the odometry at the UGV-PC, the UGV-PC requests the UGV to move following the odometry.

2.2.3 Local 5G Setup

We employed ABIT cooperation AU-510 [11] for gNB and UE, which follows the 3GPP new radio (NR) Release-15 physical layer specification. Table 2 shows the experimental setup of the local 5G in the demonstration. One antenna transmission and two antenna diversity reception are used. The transmission bandwidth is 100 MHz, and the number of resource blocks (RBs) is set to 264. The uplink and downlink slot ratio for data transmission are set to 6/20, 5/20, respectively. We fixed the modulation and coding scheme (MCS) index of 22 [12], i.e., 64QAM modulation and coding rate of approximately 0.65. As a result, the uplink and downlink data rate of 73.8 Mbps, 88.5 Mbps, respectively.

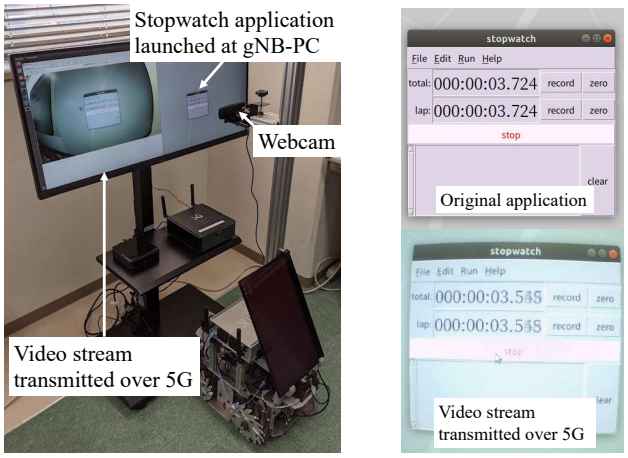
Table 3: Application-level delay for different parameters.

Resolution	Compression	Delay	Throughput	
			Uplink	Downlink
320×240	Not applied	128 ms	57.8 Mbps	1.07 Mbps
320×240	Applied	112 ms	1.9 Mbps	0.03 Mbps
1980×1020	Not applied	1318 ms	87.2 Mbps	1.57 Mbps
1980×1020	Applied	168 ms	30.5 Mbps	0.54 Mbps

2.3 Delay Performance

The delay performance is very important for the demonstration. We first evaluate the application-level delay performance. Figure 5 shows the application-level delay evaluation setup and its results. In order to evaluate the application delay in a simple way, the stopwatch application is launched at the gNB-PC. The webcam connected to the UGV-PC captures the stopwatch application displayed on the gNB-PC monitor, and the UGV-PC sends this video streams to the gNB-PC via the local 5G uplink. By taking the screenshot at the gNB-PC, we can estimate the application-level delay. Figure 5(b) shows the results when we set the video stream of webcam with 1920×1080 resolution at 30 fps. In this case, the application-level delay is approximately 0.17 seconds. Since the value of the delay is highly dependent on the timing at which the screenshot is taken, we took 10 shots and obtained the average value of the delay. Table 3 shows the results obtained for different resolutions, with and without compression applied. In the table, the uplink and downlink throughput as well as delay is presented. By setting the resolution of 320×240 at 30 fps, the data rate becomes $320 \times 240 \times 24 \times 30 \approx 55.2$ Mbps (24-bit color is assumed), which roughly coincides with the uplink throughput considering the header insertion overhead. By applying the compression, the delay decreases slightly. This may be due to the fact that the effect of reducing the transmission time of one frame of video stream in the wireless part by reducing the amount of data is larger than the increase in the processing time of compression slightly. Furthermore, when the high resolution video stream is transmitted without data compression, the data rate becomes higher than the local 5G uplink. As a result, the delay becomes much increased. However, by applying the compression, the data rate of the video stream is lower than that of the local 5G uplink, and small delay performance is achieved. Note that the downlink throughput is relatively small, since the main traffic is the control information on the control pad.

In order to clarify the proportion of the application-level delay that is occupied by the delay in the wireless part, Figure 6 shows the round trip delay (RTT) performance between gNB and UE. As shown in the figure, the maximum / average / minimum delay is



(a) Setup. (b) Results.

Fig. 5: Application-level delay performance.

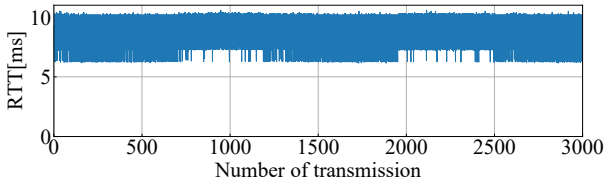


Fig. 6: Wireless delay performance (`ping`).

6.15 / 8.482 / 10.694 ms, which is less than a tenth of the application-level delay. Note that the RTT includes both uplink and downlink, while the application-level delay only considers the uplink. As a result, the wireless part is not considered to be a major factor in the delay.

2.4 Demonstration at Kagawa University

We performed the small-scale demonstration at the Hayashicho campus of Kagawa university, which is located at Takamatsu city in Japan. Figure 7 schematically illustrates the demonstration site. As shown in the figure, the gNB was installed near the window on the side of the lecture room on the third floor of the building. A maze of 3 m × 3 m was set up on the landing outside the lecture room. As shown in the figure, the maze is located about 10 m from the gNB, and non line-of-sight (NLOS) condition. Therefore, the operator needs to control the UGV from the webcam view and 2D map and location presented on the monitor. Although the door is basically open so that people can freely enter and leave the lecture room, wireless communication quality is not a problem even when the door is closed. Although a limited number of people joined the demonstration due to the COVID-19 situation, they enjoyed the demonstration.

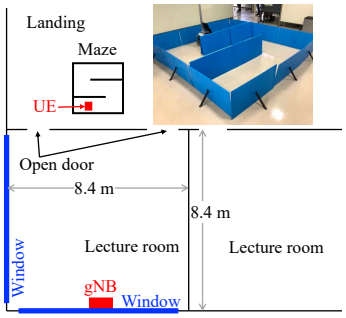


Fig. 7: Demonstration environment.

3. Conclusion

We built up the small-scale system which demonstrates the remote control of a UGV using local 5G system, and evaluated the delay performance. The application-level one way delay over the small-scale system is approximately 170 ms, which is measured by the webcam application. Furthermore, the `ping` RTT is approximately 8.5 ms. Therefore, we conclude that the wireless part is not considered to be a major factor in the delay in this demonstration. Finally, we used the small-scale demonstration system at the campus tour for high school students. Although there were some limitations due to COVID-19, we gave students an opportunity to experience local 5G.

References

[1] Ericsson, "Ericsson mobility report," Nov. 2020.
[2] Qualcomm, "Private LTE networks," Jul. 2017.
[3] 3GPP, "Service requirements for cyber-physical control applications in vertical domains v16.5.0," Jul. 2020.
[4] 5GMF, "General Report on 5G System Trials in Japan from 2017 to 2020," Mar. 2021.
[5] HOKUYO AUTOMATIC CO., LTD., "Scanning rangefinder URG-04LX-UG01." [Online]. Available: https://www.hokuyo-aut.jp/search/single.php?serial=166
[6] M. Quigley, K. Conley, B. P. Gerkey, J. Faust, T. Foote, J. Leibs, R. Wheeler, and A. Y. Ng, "ROS: an open-source Robot Operating System," *ICRA Workshop on Open Source Software*, 2009.
[7] Vstone Co.,Ltd., "Mecanumrover version 2.1." [Online]. Available: https://www.vstone.co.jp/products/wheelrobot/ver2.1.html
[8] A. L. Ballardini, S. Fontana, A. Furlan, and D. G. Sorrenti, "ira_laser_tools: a ROS laserscan manipulation toolbox," *CoRR*, vol. abs/1411.1086, 2014. [Online]. Available: http://arxiv.org/abs/1411.1086
[9] H. Durrant-Whyte and T. Bailey, "Simultaneous localization and mapping: part I," *IEEE Robotics Automation Magazine*, vol. 13, no. 2, pp. 99–110, 2006.
[10] T. Bailey and H. Durrant-Whyte, "Simultaneous localization and mapping (SLAM): part II," *IEEE Robotics Automation Magazine*, vol. 13, no. 3, pp. 108–117, 2006.
[11] ABIT Corporation, "AU-510." [Online]. Available: https://www.abit.co.jp/products/au-500/
[12] 3GPP, "NR; Physical layer procedures for data V15.0.0," Dec. 2017.