Effect of QoS Control in Remote Master-Slave Robot Systems with Force Feedback

Nuzrath Hameedha A and Yutaka Ishibashi
Graduate School of Engineering, Nagoya Institute of Technology, Nagoya, Japan
Email: {m.abdul.368}@stn.nitech.ac.jp, {ishibasi}@nitech.ac.jp

Konstantinos E. Psannis
Department of Applied Informatics, University of Macedonia, Thessaloniki, Greece
Email: {kpsannis}@uom.edu.gr

***Abstract*—In this paper, we compare effects of three types of QoS (Quality of Service) control for cooperative work in remote master-slave robot systems with force feedback. Two of the three types are the adaptive Δ-causality control and the robot position control with force information, which we previously proposed. The remaining one is a combination of the two types of control. In remote master-slave systems with force feedback, one remote robot system acts as a master and the other system acts as a slave. A user of the master system can remotely operate a robot of the master system by using a haptic interface device. A robot of the slave system automatically follows the robot of the master system. We perform cooperative work of moving a wooden stick held by the robot arms of the master and slave systems. Experimental results show that the combination of the two types of control is the most effective.**

***Index Terms*-remote robot system, master-slave systems, force feedback, cooperative work, QoS control, network delay**

I. Introduction

In the recent years, research interests in remote robot systems with force feedback have been increased exponentially [1]-[4]. In a remote robot system with force feedback, a user remotely controls a robot by using a haptic interface device while watching video. With multiple remote robot systems with force feedback [5], the efficiency and accuracy of cooperative work can largely be improved. Proper communication between the systems plays an integral role in achieving smooth completion of the work. However, when the force information is transmitted over a QoS (Quality of Service) non-guaranteed network such as the Internet, unfavorable network conditions such as network delay, delay jitter, and packet loss result in serious degradation of QoS [6] and QoE(Quality of Experience) [7]. Also, instability phenomena in the systems may largely affect the remote operation. To solve the problems, it is necessary to carry out stabilization control and QoS control [8]-[10]. In this paper, we focus on QoS control.

A number of excellent QoS control techniques for the remote robot systems with force feedback has been proposed in the recent years. In [10], the authors have investigated effects of the adaptive Δ-causality control [please refer to the original paper] in remote robot systems with master-slave relation called the *remote master-slave robot systems* here. They show that the influence of network delay between the remote master and slave systems has been significantly mitigated by using the control. However, the effect of the control for the network delay in each remote robot system has not been clarified quantitatively.

In [11], the robot position control with force information is proposed to reduce large force that is applied to an object by adjusting the robot position finely in the direction to reduce the force in cooperative work for the case of equal relation between robots. In [12], the robot position control with force information is enhanced by analyzing the relation between the length of the object and the force applied to the object. However, the effect of the control has not been verified for the remote master-slave robot systems.

In this paper, we compare effects of the adaptive Δ-causality control and robot position control with information in the remote master-slave robot systems. We also examine the effect of a combination of the adaptive Δ-causality control and the robot position control with force information in the systems to clarify which type of QoS control is the most effective by experiment.

The remainder of this paper is organized as follows. Section II explains about the remote master-slave robot systems, and Section III describes the two types of QoS control. Section IV explains the experimental method, and experimental results are presented in Section V. Section VI concludes the paper.

II. Remote Master-Slave Robot Systems

A. System Configuration

The configuration of the remote master-slave robot systems is shown in Fig. 1. The experimental setup consists of the two remote robot systems (called the *master and slave terminals* here). Each system consists of a master terminal and a slave terminal. The master terminal is composed of PC for haptic interface device

Manuscript received August 11, 2020; revised January 7, 2021.

doi: 10.18178/ijmerr.10.2.49-53

and PC for video. The former PC has a haptic interface device (3D Systems Touch [13]). The two PCs are connected to each other by using a switching hub. The slave terminal consists of PC for industrial robot and PC for video, which are connected to each other by using a switching hub. PC for industrial robot is directly connected to an industrial robot via an Ethernet (100BASE-TX) cable. The industrial robot consists of a robot arm (RV-2F-Dby Mitsubishi Electric Corp.), a robot controller (CR750-Q), a force interface unit (2F-TZ561), a force sensor (1F-FS001-W200) which is attached to the robot arm. The robot arm of each system has a toggle clamp hand (see Fig. 2).

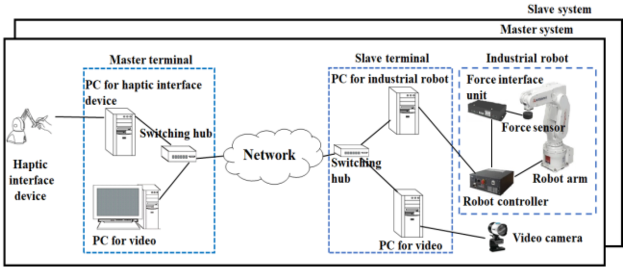


Figure 1. Configuration of remote master-slave robot systems.

B. Master-Slave Relation

In each system, a user can remotely operate the industrial robot at the slave terminal by using the haptic interface device at the master terminal, while watching video. In the remote master-slave robot systems, a user of the master system uses the haptic interface device to move both robots cooperatively while watching video. In the system having the master robot, PC for haptic interface device of the master terminal sends the position information to PC for industrial robot. Then, the position information is transmitted from PC of the master system to PC of the slave system, thereby enabling cooperative movement using the master-slave relation. Each robot employs the real-time control function and real-time monitor function [14] to get position and force information of its robot arm. The force information is transmitted from PC for industrial robot to PC for haptic interface device in each system. A user of the slave system can perceive the reaction force by holding the haptic interface device of the slave system.

III. QoS Control

In this section, we describe the two types of QoS control: The adaptive Δ-causality control (referred as *ADC* here) and the robot position control with force information (called *RPC*).

A. Adaptive Δ-causality Control

When there is a large network delay between the master system and slave system, the position information of the robot of the slave system lags behind that of the master system. This difference in position results in large force applied to an object carried by the robots. In the adaptive Δ-causality control [10], the output timing of the master robot position information is delayed dynamically according to the network delay so that both the master and slave robots move at the same time. The output timing of the position information at the master robot is set to the generation time + Δ. The value of Δ is changed dynamically according to the network delay. Let us denote the value of Δ at time t by Δ_t here. The value of Δ_t is obtained by smoothing the network delay d_t measured at time t by the following equation:

$$\Delta_t = \alpha\Delta_{t-1} + (1 - \alpha)\, d_t. \quad (1)$$

where α is a smoothing coefficient and is set to 0.998 [10].

B. Robot Position Control with Force Information

The robot position control with force information finely adjusts the robot position according the force applied to the object carried by the robot arms. In [12],the authors obtain the relation between the force and position information as a function of the length of the stick that is carried by the robot arms. The new position information $\hat{\boldsymbol{S}}_t$ is obtained by adding $\boldsymbol{P}$ to position St as follows:

$$\hat{\boldsymbol{S}}_t = \boldsymbol{S}_t + \boldsymbol{P}. \quad (2)$$

The robot arm is moved $\boldsymbol{P}$ in the direction so as to reduce the large force applied to the stick. In this paper, the robot position control with force information is applied to only the slave system. This is because applying the robot position control with force information in one system yields better results than in the two systems [15].

IV. Experimental Method

In the experiment, we carried out a task of touching paper blocks with a wooden stick grasped by the two robot arms. The wooden building blocks are piled-up before and behind the wooden stick, and a paper block is placed on each upper most building block as shown in Fig. 2. The task is to touch each paper blocks (front and back). The two paper blocks are placed at 80 mm from each other. The initial position of the wooden stick is set so that it is at an equal distance from both the paper blocks (i.e., 40 mm from each paper block). We touched the paper block on the front side in 5 seconds and that on the back side in 10 seconds (i.e., it took about 15 seconds to complete the task). The force mapping ratio between the haptic interface device and the robot arm in each system is set to 1:2 [5]. The robot arm of each system is allowed to move in front and back direction. The movement in left and right, and up and down directions is restricted for simplicity.

In the remote master-slave robot systems, the master and slave systems are connected via a network emulator [16], and a constant delay was added to each packet sent between the two robots, and between the master and slave terminals of each system (the one-way constant delay is called the *additional delay* here).The experiment was carried out for different network delays between the two PCs for industrial robots, and between master and slave terminals of master system. The additional delay between the master and slave robots (i.e., the two slave terminals) was varied from 0 ms to 200 ms at intervals of 50 ms. The additional delay between the master and slave terminals of the master system was varied from 0 to 200 at intervals of 100 ms. It should be noted that the additional delay between the master and slave terminals of the slave system was always set to 0 ms in this paper because the additional delay does not affect the task. The task was repeated 10 times for each combination of additional delays.

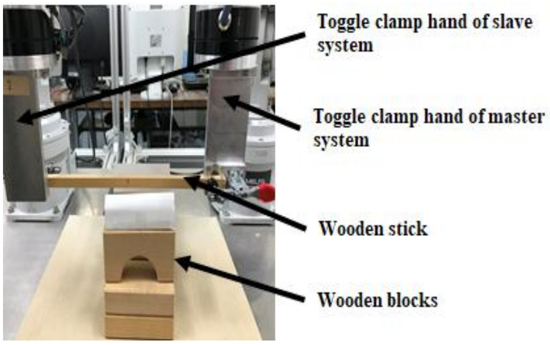
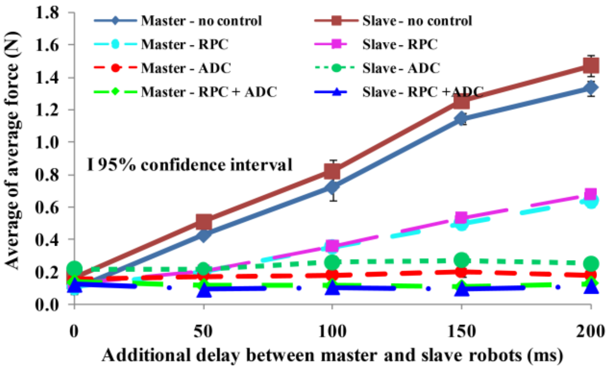
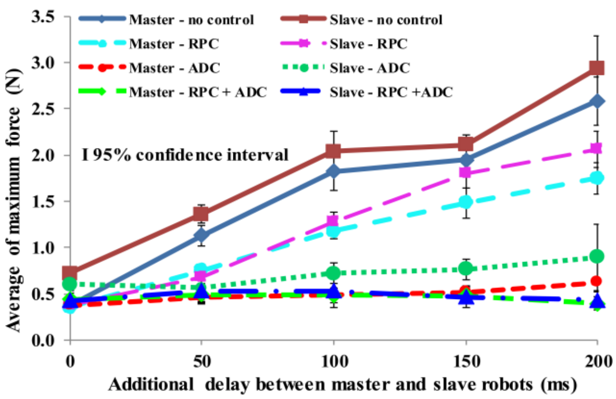


Figure 2. Arrangement of paper and wooden blocks.

To compare effects of different types of QoS control, we use the average of average force and the average of maximum force at each robot as performance measures. The average of the average force at each robot is defined as the 10 times average of the temporal average force during each task at each robot. The average of maximum force is obtained by averaging the maximum force during each task at each robot.



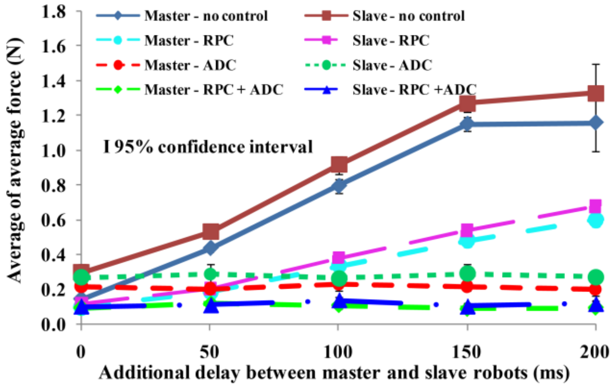
(a) Average of average force



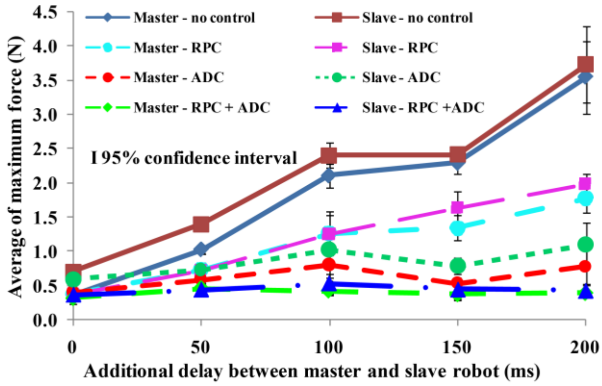
(b) Average of maximum force

Figure 3. Average of average force and average of maximum force for additional delay of 0 ms in master system.

V. Experimental Results



(a) Average of average force



(b)Average of maximum force

Figure 4.Average of average force and average of maximum force for additional delay of 200 ms in master system.

We show the average of average force and the average of maximum force versus the additional delay between the master and slave robots for additional delays of 0 ms and 200 ms between the master and slave terminals of the master system in Figs. 3 and 4, respectively. In the figures, the 95% confidence intervals are also included. Furthermore, *RPC* and *ADC* stand for the robot position control with force information and the adaptive Δ-causality control, respectively. RPC + ADC means a combination of RPC and ADC. *No control* does not carry out any QoS control.

In Figs. 3 and 4, we see that the average of average force and the average of maximum force for additional delays of 0 ms and 200 ms between the master and slave terminals of the master system are almost the same. This means that the average of average force and the average of maximum force hardly depends on the additional delay between the master and slave terminals of the master system.

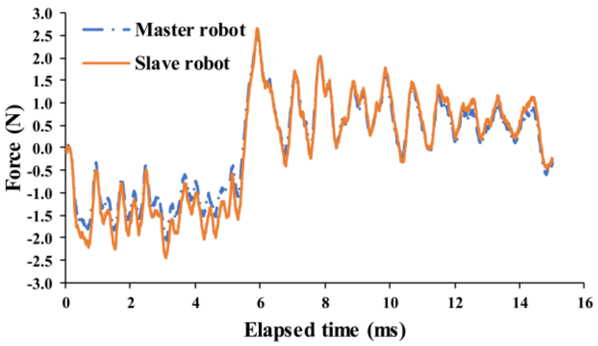
We also see in the figures that the average of average force and the average of maximum force for RPC + ADC are the smallest. ADC has the second smallest, RPC has the third smallest, and no control has the largest. Therefore, ADC has larger effect than RPC.

We further observe in the figures that the slave system tends to have larger average of average force and average of maximum force than the master system. We are now clarifying the reasons; we will describe the reasons in the camera-ready version of this paper (one of the reasons may be the difference in direction of the toggle clamp hand setup between the two robots).

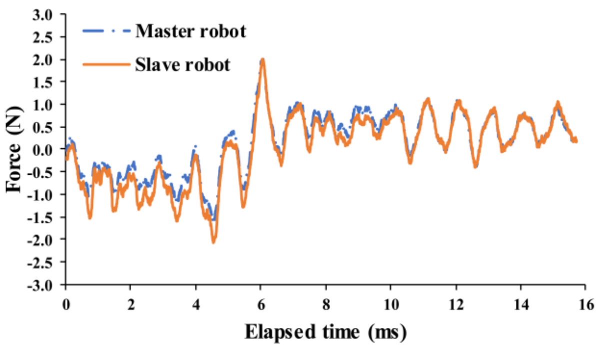
To examine the effects of different types of QoS control more clearly, we show the force of the master and slave robots as a function of the elapsed time from the beginning of each task in Fig. 5, where we set the additional delay between the master and slave robots that between master and slave terminals of the master system to 200 ms. Figs. 5 (a) through (d) plot the results for no control, RPC, ADC, and RPC + ADC, respectively. The results for the other combinations of additional delays are not shown here because the results are almost the same as those in Fig. 5.

In Figs. 5 (a) and (b), we observe that the force of no control fluctuates greatly, while the fluctuations are reduced when RPC is used. We also notice that the force jumps up and the sign of the force is also reversed at about 6 seconds. The reason is that the direction of movement to touch the paper block on the back side after touching the paper block on the front side is changed at about 5 second as mentioned in Section Ⅳ.On the other hand, we can see that the force of the robots hardly increases with the additional delay in Figs. 5 (c) and (d). We note that force of the robots for RPC + ADC is smaller than that of ADC.

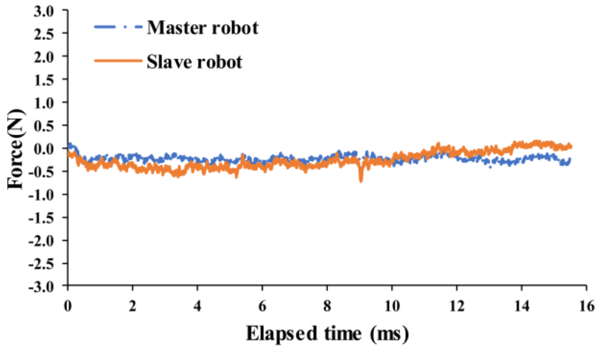
From the above considerations, we can confirm that the combination of the robot position control with force information and the adaptive Δ-causality control is the most effective.



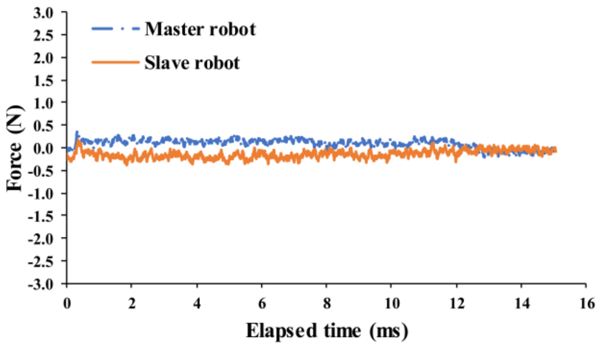
(a) No control



(b) RPC



(c) ADC



(d) RPC + ADC

Figure 5. Force versus elapsed time (additional delay: 200 ms).

VI. CONCLUSIONS

In this paper, we made a comparison of the following three types of QoS control: The adaptive Δ-causality control, the robot position control with force information, and a combination of both. As a result, we found that the combination of the adaptive Δ-causality control and the robot position control with force information is the most effective. We also confirmed that force fluctuations are alleviated greatly when compared to the other two types of QoS control. Furthermore, the effect of the adaptive Δ-causality control is larger than that of the robot position control with force information

As our future work, we will apply the combination of the adaptive Δ-causality control and the robot position control with force information to other types of cooperative work.

CONFLICT OF INTEREST

The authors declare no conflict of interest.

AUTHOR CONTRIBUTIONS

All the authors conducted the research; NuzrathHameedha A analyzed the data and wrote the paper, Yutaka Ishibashi, and Konstantinos E. Psannis revised the paper; all the authors had approved the final version.

ACKNOWLEDGEMENT

This work is supported by JSPS KAKENHI Grant Number 18K11261.

REFERENCES

[1] P. Nadrag, L. Temzi, H. Arioui, and P. Hoppenot, "Remote control of an assistive robot using force feedback," in *Proc. The 15th International Conference on Advanced Robotics (ICAR)*, June 2011, pp. 211-216.

[2] E. Steinbach, S. Hirche, M. Emst, F. Brandi, R. Chaudhari, J. Kammerl and I. Vittorias, "Haptic communications," *IEEE Journals & Magazines*, vol. 100, no. 4, Apr. 2012, pp. 937-945.

[3] T. Miyoshi, Y. Maeda, Y. Morita, Y. Ishibashi, and K. Terashima, "Development of haptic network game based on multi-lateral tele-control theory and influence of network delay on QoE," (in Japanese), *Trans. the Virtual Reality Soceity of Japan, Special Issues on Haptic Contents*, vol.19, no. 4, Dec. 2014, pp. 559-569.

[4] Y. Ishibashiand P. Huang, "Improvement of QoS in haptic communication and its future," *IEICE Trans. Communication* (in Japanese), vol. J99-B, no. 10, Oct. 2016, pp. 16-20.

[5] E. Taguchi, Y. Ishibashi, P. Huang, and Y. Tateiwa, "Experiment on collaborative work between remote robot systems with haptics," (in Japanese), *IEICE Global Conference*, B-11-17, Mar. 2018.

[6] ITU-T Rec. I. 350, "General aspects of quality of service and network performance in digital networks," Mar. 1993.

[7] ITU-T Rec. G. 100/P. 10 Amendment 1, "New appendix I – Definition of quality of experience (QoE)," 2007.

[8] P. Huang and Y. Ishibashi, "QoS control in haptic communications," (in Japanese), *Journal of IEICE, Special Section: Advancement and Quality Enhancement in Haptic Communications*, vol. 102, no. 1, Jan. 2019, pp.42-46.

[9] K. Suzuki, Y. Maeda, Y. Ishibashi, and N. Fukushima, "Improvement of operability in remote robot control with force feedback," in *Proc. IEEE Global Conference on Consumer Electronics (GCCE)*, Oct. 2015, pp.16 -20.

[10] K. Kanaishi, Y. Ishibashi, P. Huang, and Y. Tateiwa, "Effects of adaptive Δ-causality control for cooperation between remote robot systems with force feedback by using master-slave relation," in *Proc. The 3rd International Conference on Telecommunication and Communication Engineering (ICTCE)*, Nov. 2019.

[11] Y. Ishibashi, E. Taguchi, P. Huang, and Y. Tateiwa, "Robot position control with force information in cooperation between remote robot system," in *Proc. IEEE The5thInternational Conference on Control, Automation and Robotics (ICCAR)*, Apr. 2019.

[12] S. Ishikawa, Y. Ishibashi, P. Huang, and Y. Tateiwa, "Effect of robot position control with force information for cooperative work between remote robot systems," in *Proc. The 2nd World Symposium on Communication Engineering (WSCE)*, Dec. 2019, pp. 210-214.

[13] 3D Systems Touch, [Online]. Available:https://www.3dsystems.cpm/haptic-devices/touch.

[14] RV-2F-D Series: Standard Specifications, (in Japanese), [Online]. Available: http://www.mitsubishielectric.co.jp/dl/fa/members/document/manual/robot/bfp-a8899/ bfp-a8899u.pdf.

[15] S. Ishikawa, Y. Ishibashi, P. Huang, and Y. Tateiwa, "Effects of robot position control using force information in remote robot systems with force feedback: Comparison between human-robot and robot-robot cases," in *Proc. The 2nd International Conference on Computer Communication and the Internet (ICCCI)*, June 2020.

[16] M. Carsonand D. Santay, "NIST Net: a Linux-based network emulation tool," *ACM SIGCOMM Computer Communication Review*, vol. 33, no. 3, July 2003, pp. 111-126.

Nuzrath Hameedha A received the B.E., and M.E., degrees from Anna University,India, in 2010, and 2013, respectively. She is currently working towards her Ph.D. in the Department of Computer Science, Graduate School of Engineering, Nagoya Institute of Technology. Her research interests include networked multimedia, QoS (Quality of Service) control and QoE (Quality of Experience) assessment in remote robot system with force feedback. She is a student member of IEEE.

Yutaka Ishibashi received the B.E., M.E., nadDr.E. degrees from Nagoya Institute of Technology, Japan in 1981, 1983, and 1990, respectively. In 1983, he joined the Musashino Electrical Communication Laboratory of NTT. From 1993 to 2001, he served as an Associate Professor of Department of Electrical and Computer Engineering, Faculty of Engineering, Nagoya Institute of Technology. Currently, he is a Professor of Department of Computer Science, Graduate School of Engineering, Nagoya Institute of Technology. His research interests include networked multimedia, QoS control, media synchronization, and remote robot control. He is a fellow of IEICE, a senior member of IEEE, and a member of ACM, ITE, IPSJ, and VRSJ.

Konstantinos E. Psannisreceived the degree in Physics from Aristotle University of Thessaloniki, Greece, and the Ph.D. degree from the Department of Electronic and Computer Engineering, School of Engineering and Design, Brunel University, London, U.K. He is currently an Associate Professor in communications systems and networking with the Department of Applied Informatics, School of Information Sciences, University of Macedonia, Greece, and the Director of the Mobility2net Laboratory, JAPAN-EU Research and Development and Consulting. His research interests include digital media communications, media coding/synchronization, and transport over a variety of networks. He serves as an Associate Editor for IEEE ACCESS and the IEEE COMMUNICATIONS LETTERS.