SUGISAKA MASANORI

Journal of Advances in Artificial Life Robotics
Vol. 3(4); March (2023), pp. 209–213
ON LINE ISSN 2435-8061; ISSN-L 2435-8061
https://alife-robotics.org/jallr.html

Research Article

QoS Balancing Optimization in Aggregated Robot Processing Architecture: Rate and Buffer

Abdul Jalil, Jun Kobayashi, Takeshi Saitoh

Kyushu Institute of Technology, 680-4 Kawazu, Iizuka-shi, Fukuoka, 820-8502, Japan

ARTICLE INFO

Article History

Received 25 November 2022
Accepted 30 May 2023

Keywords

QoS
Optimization
Aggregated robot processing
ROS 2

ABSTRACT

This study developed an optimization algorithm to balance the Quality of Service (QoS) rates and buffer size for robot data communication in the Aggregated Robot Processing (ARP) architecture. Robot Operating System 2 (ROS 2) is robotic software that uses a set of QoS policies to control the quality of data transmission in robotic networks, such as DEADLINE to determine the rates, and DEPTH to determine the buffer size. An unbalanced DEADLINE and DEPTH configuration in ROS 2 node communication can result in high packet latency and packet loss in RELIABLE connections. This happens when the DEADLINE sets the rates at high frequencies, and the DEPTH sets the buffers with small sizes. The results of this study show that the optimization algorithm developed in this study can determine the rate and buffer size through a balancing configuration for better quality robot data transmission in ARP architecture, influence the improvement of latency and reduce the packet loss.

1. Introduction

The data processing flow in robotic systems commonly consists of sensing, planning, and actuation components [1]. In the network, each component of these data processing flows can communicate like a node and manage the quality of data communication using Quality of Service (QoS) policies. The purpose of QoS is to manage the data communication systems to reduce some problems in the network, such as latency, packet loss, and jitter. However, suppose that QoS configures the data traffic in the network with an unbalanced configuration. In that case, the quality of data transmission will be reduced, resulting in high data transfer latency and packet loss.

In this study, we developed an optimization algorithm to determine QoS rates and buffer size with balancing configuration when ROS 2 is used to transmit robot data in the ARP architecture. ROS 2 is a robotic software developed above DDS (Data Distribution Service), then implements QoS to tune the quality of the data transmission system between the publisher and the subscriber [2]. ROS 2 has some QoS configurations to tune the performance of robot data transfer in the network, such as DEADLINE and DEPTH. DEADLINE is a QoS policy to determine data transfer rates in the network, and then DEPTH configures the buffer size used to store the data sample in publish-subscribe communication.

The study on analyzing the efficiency of QoS policies configuration in ROS 2 node communication is an interesting topic that some researchers have conducted. They analyzed the latency of data transmission [2][3][4][5][6], throughput performance [2][4], packet data loss in the network [4][5][6][7], and memory consumption in the robot computer [2]. Another study also analyzed when the QoS policy configured DEADLINE with high rates and DEPTH with a small buffer, and the results show that some packets were lost in the communication of the node [4][5][6][7]. Otherwise, suppose that DEPTH sets the buffer with a large size, and DEADLINE sets the rates with low frequency. Memory consumption increases in that configuration [4][8], and affects real-time data transfer between the publisher and the subscriber.

Corresponding author's E-mail: malla.abdul-jalil545@mail.kyutech.jp, jkoba@ics.kyutech.ac.jp, saitoh@ai.kyutech.ac.jp URL: www.kyutech.ac.jp

This paper contributes a study on improving robot data transfer in the ARP architecture by balancing the QoS DEADLINE and DEPTH configuration when the ROS 2 node communication implements the RELIABLE connection and KEEP_LAST option to send the data sample from publisher to subscriber. We analyze it because strict reliability is not achieved if the RELIABLE connection uses the KEEP_LAST option in the publish-subscribe communication [9].

The chapter in this paper consists of four parts. In the second chapter, we elaborate on the proposed optimization algorithm used in this study to balance the DEADLINE and DEPTH QoS configuration. In the third chapter, we show the results of this study and our conclusions in the fourth chapter.

2. Methods

In the ARP, the robot computer function runs the sensing and actuation component, and the CEDDP runs the planning component. In the sensing component, the robot will manage the sensor hardware, read the sensor output, and send the sensor data to the planning component through a wireless network. Furthermore, in the planning component, the CEDDP processes the sensor data to determine the robot's localization and path planning. After that, the actuation component controls the robot's actuator moving according to the localization result sent from the planning component through a wireless network. Fig.1 shows the architecture of the ARP proposed in our study.

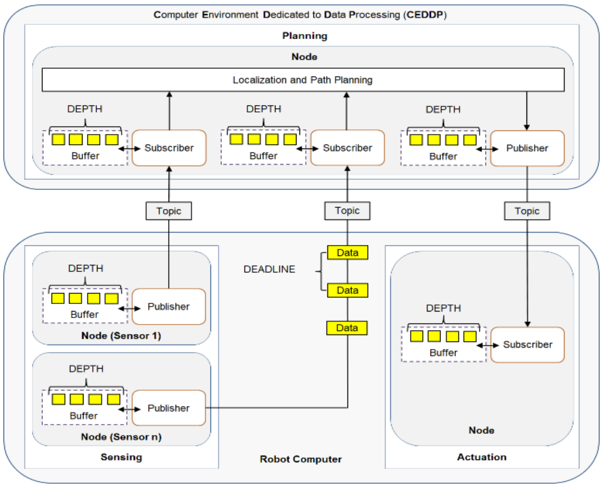

Fig.1.ARP architecture [10].

According to the information in Fig.1, to determine the optimal value of the DEADLINE and DEPTH, first, we need to calculate the number of topics used to communicate the data sample between the sensing to the planning component and from the planning to the actuation component. In this study, to determine the optimal value of the DEADLINE, we divide the maximum data transmission rate by the number of topics used in the ARP architecture.

$$R = \frac{Rmax}{T} \quad (1)$$

Where R is the rate, $Rmax$ is the maximum rate for transferring the data in a topic, and T is the number of topics used in ARP. Based on the formula shown in Eq. (1), we divide $Rmax$ by T to balance the data transmission rates with all topics in the node communication. So, according to our idea shown in Eq. (1), determine the first constraint with the following equation:

$$\frac{Rmax}{T} - R \geq Rmin \quad (2)$$

According to the constraint shown in Eq. (2), the variable R should be larger than or equal to the minimum rate of data transmission $Rmin$ to ensure that the nodes of each component can communicate in the ARP architecture. For the next constraint, find the optimal DEPTH value to determine the buffer size and balance it with the DEADLINE tune. In this constraint, if the DEADLINE tunes the rates with high frequency and close to the maximum rate, the DEPTH will also tune the buffer to a large size and close to the maximum buffer size. Otherwise, if the DEADLINE tunes the rates with low frequency, the buffer size configured in the DEPTH will also be small and close to the minimum buffer size. Furthermore, formulate the second constraint to balance the DEPTH and DEADLINE tune with the Eq. (3):

$$Dmax \frac{R}{Rmax} - D \geq Dmin \quad (3)$$

Where D is the buffer size, $Dmax$ is the maximum buffer size when the KEEP_LAST option is chosen (5000 [8]), then $Dmin$ is the minimum buffer size. For the following constraints, bound the variable R between the maximum and minimum rates and the variable D between the maximum and minimum buffer sizes, $Rmin \leq R \leq Rmax$ and $Dmin \leq D \leq Dmax$, respectively. For the objective function, we used multi-objective optimization [11] to determine the maximum value of DEADLINE and DEPTH. Finally, build our idea to find the optimal value of DEADLINE R and DEPTH D with optimization:

$$\max R, D$$
$$s.t. \quad \frac{Rmax}{T} - R \geq Rmin$$

$$Dmax \frac{R}{Rmax} - D \geq Dmin \quad (4)$$
$$Rmin \leq R \leq Rmax$$
$$Dmin \leq D \leq Dmax$$

To implement the optimization, CVXPY was used in this study to solve the problem. CVXPY is an open-source Python-embedded modeling language used to solves the convex optimization problem [12]. Furthermore, the following is an algorithm developed to implement our optimization created in Eq. (4).

Algorithm: Optimization algorithm

Require: $\boldsymbol{T}, \boldsymbol{Rmax}, \boldsymbol{Rmin}, \boldsymbol{Dmax}, \boldsymbol{Dmin}$
Variable: $\boldsymbol{R}, \boldsymbol{D}$
Constraints:
$$[\frac{\boldsymbol{Rmax}}{\boldsymbol{T}} - \boldsymbol{R} \geq \boldsymbol{Rmin},$$
$$\boldsymbol{Dmax} \frac{\boldsymbol{R}}{\boldsymbol{Rmax}} - \boldsymbol{D} \geq \boldsymbol{Dmin},$$
$$\boldsymbol{Rmin} \leq \boldsymbol{R} \leq \boldsymbol{Rmax},$$
$$\boldsymbol{Dmin} \leq \boldsymbol{D} \leq \boldsymbol{Dmax}]$$
Objective_Function: $\boldsymbol{max\ R, D}$
Problem: (Objective_Function, Constraints)
Problem.solve()

3. Results and Discussion

To perform the experiment, for the robot computer, we used a Raspberry Pi 4 with an ARM Cortex-A72 processor @ 1.5GHz and 8GB of memory. Then the laptop computer for CEDDP with an Intel Core i5 processor @ 2.60GHz and memory 12GB. We used an access point that has a frequency of 2.4 GHz to communicate the robot computer with CEDDP. Furthermore, the software installed on the robot computer and CEDDP are as follows: Linux Ubuntu 20.04 for the OS, ROS 2 Foxy Fitzroy, and Fast-RTPS DDS. Fig.2 shows the experimental setup and tools used in our experiment.

Furthermore, Table 1 shows the QoS configuration used to perform the experiment in our study. According to that QoS configuration, we conducted the experiment to communicate the robot data between the sensing, planning, and actuation components according to the illustration experiment presented in Fig.3 and the optimization results are shown in Table 2. In this study, we analyze the effectively of QoS balancing optimization when the nodes in the sensing, planning, and actuation components transmit the data sample with a size of 10 bytes, 100 bytes, and 1000 bytes, respectively.

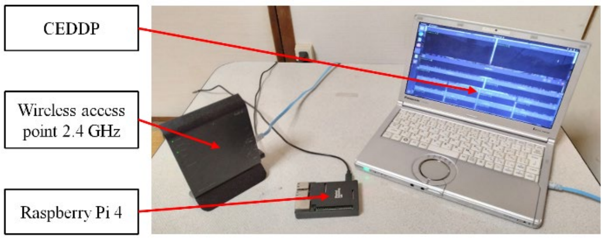

Fig.2. Tools and experimental setup in experiment.

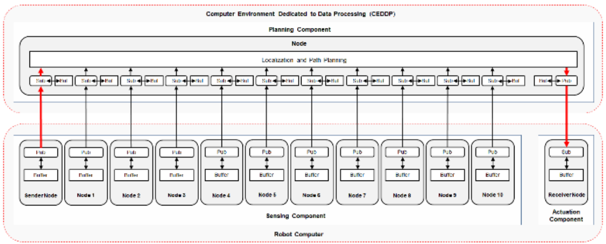

Fig.3. The illustration in experiment.

Table 1. QoS settings in experiment.

RELIABILITY	RELIABLE
HISTORY	KEEP_LAST
DEPTH	1, 5000, Optimization (D)
DEADLINE	100 Hz, 200 Hz, 500 Hz, 1000 Hz, Optimization (R)
DURABILITY	VOLATILE
LIVELINESS	AUTOMATIC

Table 2. The optimization results.

	1st Exp	**2nd Exp**	**3rd Exp**	**4th Exp**
T	12	12	12	12
$Rmax$	100 Hz	200 Hz	500 Hz	1000 Hz
$Rmin$	1 Hz	1 Hz	1 Hz	1 Hz
$Dmax$	5000	5000	5000	5000
$Dmin$	1	1	1	1
Opt (R)	**7 Hz**	**15 Hz**	**40 Hz**	**82 Hz**
Opt (D)	**365**	**390**	**405**	**410**

According to the experimental illustration presented in Fig.3 and the optimization result shown in Table 2, T = 12 is the number of topics used to communicate the data sample in the experiment. Next, the $Rmax$ value = 100 Hz, 200 Hz, 500 Hz, and 1000 Hz are the maximum rates when a topic is used to send the data sample in ARP architecture. Then the $Rmin$ value = 1 Hz is the minimum data transmission rate determined in the experiment, the $Dmax$ value = 5000 is the maximum buffer size, and the $Dmin$ value = 1 is the minimum buffer size. Furthermore, Opt (R) and Opt (D) are the results of the optimization output to determine the rates and buffer size with balancing configuration.

In this research, we evaluate the effectiveness of optimization by measuring latency and calculating packet loss in data communication between the sensing, planning, and actuation components. To analyze the latency, we calculated the time elapsed when the receiver node in the actuation component received the data sample sent from the sender node through the planning component. In Fig.4, we show the latency results of this experiment.

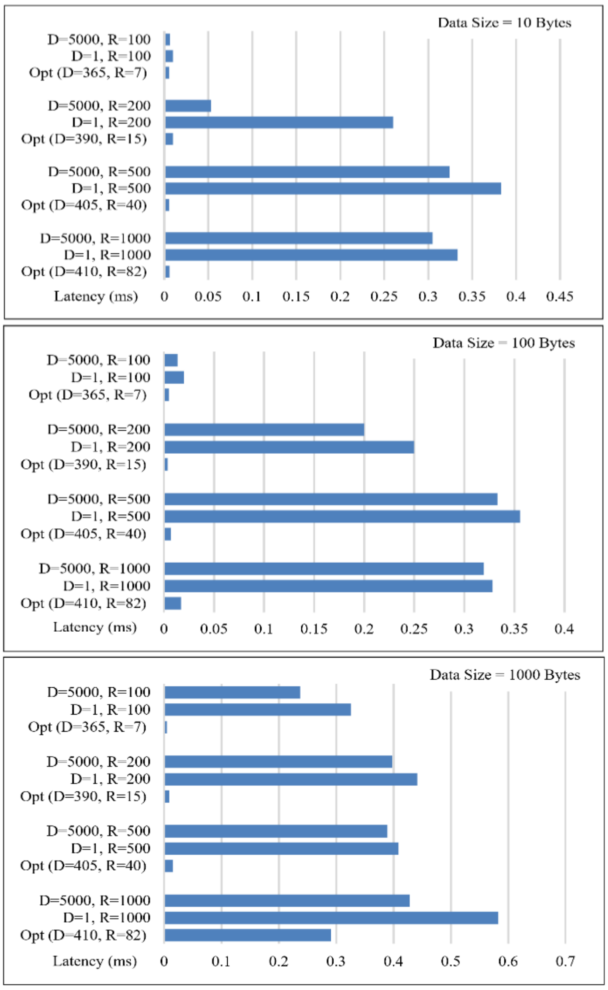

Fig.4. The results of the latency in experiment.

According to the latency results shown in Fig.4, it can be seen that the configuration of the DEADLINE and DEPTH with a balancing tune can improve the latency of packet transmission in ARP network. Furthermore, we show the packet loss results of the experiment in Fig.5.

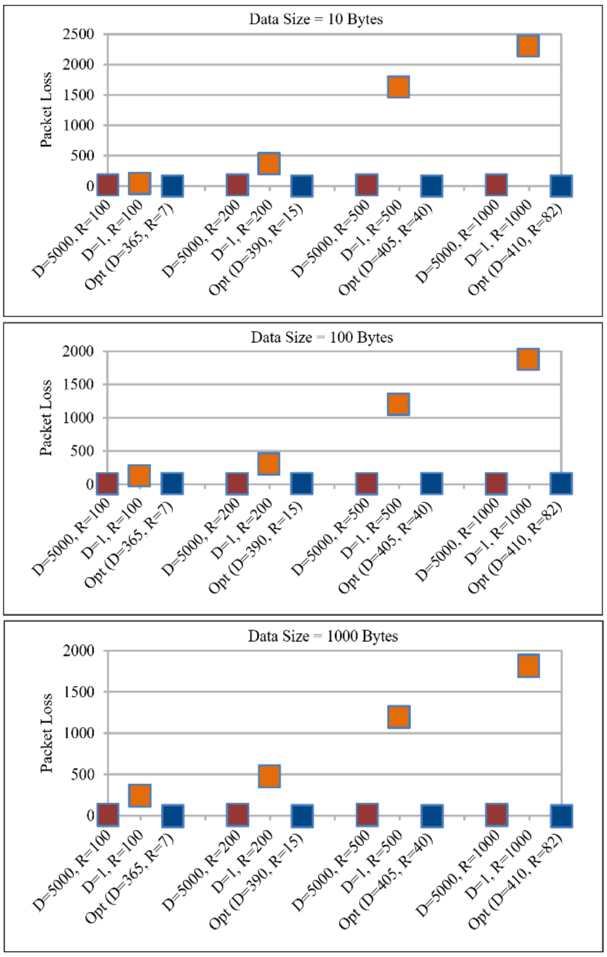

Fig.5. The results of the packet loss in experiment.

According to the analysis results presented in Fig.5, the unbalanced configuration of DEADLINE and DEPTH to determine the rates and buffer size affects the packet loss in node communication. It happens because the buffer size cannot accommodate storing the data transferred between components if the data transmission rates are too high. Furthermore, if the DEPTH sets the buffer with a large size and balances with the rates, no packet data are lost in the node communication. According to the experimental results shown in Fig.4 and Fig.5, it can be concluded that the balancing QoS configuration effectively improves the data communication systems in the ARP network.

4. Conclusion

In this study, an optimization algorithm has been developed to balance QoS configurations between rates and buffer size. Our idea in this QoS balancing optimization is to balance the DEADLINE and DEPTH QoS configuration when ROS 2 uses the RELIABLE connection and KEEP_LAST option to transmit the robot data in ARP architecture. The results of this study show that the QoS balance configuration effectively improves latency and reduces packet loss in the ARP architecture. As a result of this study, we hope to implement this QoS balancing optimization method in the next work to improve the communication and cooperation between multi-robots in the ARP architecture to enhance multi-robot data transmission.

References

1. J. Staschulat, I. Lütkebohle, and R. Lange, "The rclc Executor: Domain-specific deterministic scheduling mechanisms for ROS applications on microcontrollers: work-in-progress", International Conference on Embedded Software, EMSOFT, IEEE, 2020.
   DOI: 10.1109/EMSOFT51651.2020.9244014.
2. Y. Maruyama, S. Kato, and T. Azumi, "Exploring the Performance of ROS2", International Conference on Embedded Software, EMSOFT, IEEE, 2016.
   DOI: 10.1145/2968478.2968502.
3. T. Kronauer, J. Pohlmann, M. Matthé, T. Smejkal, and G.~Fettweis, "Latency Analysis of ROS2 Multi-Node Systems", International Conference on Multisensor Fusion and Integration for Intelligent Systems (MFI), IEEE, 2021.
   DOI: 10.1109/MFI52462.2021.9591166.
4. J. Fernandez, B. Allen, P. Thulasiraman, and B. Bingham, "Performance Study of the Robot Operating System 2 with QoS and Cyber Security Settings", International Systems Conference (SysCon), IEEE, 2020.
   DOI: 10.1109/SysCon47679.2020.9275872.
5. P. Thulasiraman, Z. Chen, B. Allen, and B. Bingham, "Evaluation of the Robot Operating System 2 in Lossy Unmanned Networks", International Systems Conference (SysCon), IEEE, 2020.
   DOI: 10.1109/SysCon47679.2020.9275849.
6. Z. Chen, "Performance Analysis of ROS 2 Networks Using Variable Quality of Service and Security Constraints for Autonomous Systems", Naval Postgraduate School, 2019.
7. J. Park, R. Delgado and B.W. Choi, "Real-Time Characteristics of ROS 2.0 in Multiagent Robot Systems: An Empirical Study", IEEE Access, vol. 8, pp. 154637-154651, 2020.
   DOI: 10.1109/ACCESS.2020.3018122.
8. eProsima, "Fast DDS Documentation", Release 2.8.1, December 2022.
9. https://community.rti.com/static/documentation/connext-dds/5.2.0/doc/manuals/connext_dds/html_files/RTI_ConnextDDS_CoreLibraries_UsersManual/Content/UsersManual/HISTORY_QosPolicy.htm.
10. A. Jalil and J. Kobayashi, "Experimental Analyses of an Efficient Aggregated Robot Processing with Cache-Control for Multi-Robot System", 20th International Conference on Control, Automation and Systems (ICCAS), IEEE, pp. 1105-1109, 2020.
    DOI: 10.23919/ICCAS50221.2020.9268225.
11. N. Gunantara, "A review of multi-objective optimization: Methods and its applications", Cogent Engineering, Vol. 5, Num. 1, pp. 1-16, 2018. DOI: 10.1080/23311916.2018.1502242.
12. S. Diamond and S. Boyd, "CVXPY: A Python-Embedded Modeling Language for Convex Optimization". Journal of Machine Learning Research, Vol. 17, Num 83, pp. 1-5, 2016.

Authors Introduction

Mr. Abdul Jalil

He received Master's and Bachelor's degree from the Department of Computer Systems, STMIK Handayani Makassar, Indonesia in 2015 and 2012, respectively. Currently, he is a Doctoral student at Department of Computer Science and Systems Engineering, Kyushu Institute of Technology, Japan.

Dr. Jun Kobayashi

He is an Associate Professor at Department of Intelligent and Control Systems, Kyushu Institute of Technology. He obtained his PhD degree at Kyushu Institute of Technology in 1999. Currently, his main interests are Robotics, Physiological Computing, and Cybernetic Training.

Prof. Takeshi Saitoh

He is Professor of Department of Artificial Intelligence, Kyushu Institute of Technology. He received B.Eng., M.Eng., and D.Eng. degrees from Toyohashi University of Technology, Japan. His research field includes image processing and pattern recognition.