Journal of Engineering Science and Technology
Special Issue, Vol. 18, No. 4 (2023) 214 - 228

PERFORMANCE OF ACIDITY NUTRIENT AUTOMATION USING MQTT PROTOCOL IN DUTCH BUCKET SYSTEM HYDROPONIC

HELMY[1*], ARIF NURSYAHID[1], THOMAS AGUNG SETYAWAN[1], ARI SRIYANTO NUGROHO[1], MUHLASAH NOVITASARI[1], SUBUH PRAMONO[2], ENI DWI WARDIHANI[1]

[1]Department of Electrical Engineering, Politeknik Negeri Semarang, JL. Prof. H. Soedarto SH, Semarang, Indonesia
[2]Department of Electrical Engineering, Univesitas Sebelas Maret, Jl. Ir. Sutarmi 36 Kentingan, Surakarta, Indonesia
*Corresponding Author: edwardihani@polines.ac.id

Abstract

The constraint of area for land agriculture ignites the practice of soilless agriculture known as hydroponic. Automation in hydroponics can help farmers to support the farming process in giving precise nutrients for the plant. One of the hydroponic techniques is the Dutch Bucket System (DBS). DBS is a hydroponic system used for taproot plants such as tomatoes. pH is the critical part of hydroponically grown plants. A pH value that is too high or too low can affect the absorption of nutrients in plant roots, so that plant growth is less than optimal. Thus, real-time pH monitoring and control has been taken into account. This paper sheds new light on hydroponic automation using edge-based and cloud computing monitoring and control with the MQTT protocol QoS level 2 which can run with or without internet connection. We conducted the statistical analysis using SPSS 25 on the work of system and QoS delay and jitter. The results of experiments with tomatoes indicate that the system automatically measures the nutrient value of pH according to the threshold determined. The control system runs very well with an average control accuracy of 98.84%. However, the Quality of Service (QoS) value in delay and jitter are categorized as bad. The bad category of delay and jitter are the consequence of MQTT protocol QoS level 2 due to a 4-way handshake to guarantee the message delivered since MQTT is designed for reliability not for speed.

Keywords: Hydroponic, pH, MQTT, QoS.

1. Introduction

A dire need for food is increasing by 2050 to feed the population from 6 to 9 billion in the world. The resources that are consumed at the same time leading to awareness of sustainable food. Hence, sustainable agriculture production is required [1]. Sustainable agriculture is a way to increase productivity in agriculture in order to boost up the level of prosperity for people [2]. In developing countries like Indonesia which is known as a country with high rainfall and tropical climate, agriculture is considered as a backbone of the economy. This is in line with agricultural contribution to GDP (Gross Domestic Product) in 2019 that ranked 3rd largest after the manufacturing and trade sector [3]. Time must be considered to meet the limited resources and the consumption demand to achieve sustainable agriculture. Therefore, the management of optimum plant nutrients in agriculture has become the priority area of research for sustainable agriculture production [4-6].

Myriad agricultural products demanded by people are various types of vegetables and fruits [7]. Meanwhile those types of farms are the power production of conventional farmers. Thus, a solution is needed for conventional farmers to survive in the middle of industrial farming [8]. One solution is to change the traditional technique of cultivation with hydroponic. Hydroponic is a soilless advanced vegetable and fruits growing technique with a solution of water and dissolved nutrient management [9].

Temperature, water level and pH level are some of the critical parameters affecting plant growth. But acidification has become a serious environmental problem in intensive agriculture. The value of how acidic the solution is measured by pH. Even if pH falls as the temperature increases, this doesn't mean that water is more acidic at higher temperature [10]. A solution is acidic when there is an excess of hydrogen ions over hydroxide ions. This is due to the excessive use of acidic and physiologically acidic nitrogen fertilizers as well as the acid rain by environmental pollution [11]. Meanwhile, among the problems in hydroponics includes water quality and pH fluctuations. Plants need water and nutrients to grow well. Good yields and high quality of crops can be achieved when water and nutrients are given in the required quantities [12]. pH plays an important role in water quality and determination of how acidic the nutrient is [13]. Basically, in the natural environment, pH plays roles as a master variable in biogeochemical processes which influences numerous biological, chemical and physical properties which affect plant growth, dry matter production, root rhizosphere and availability of nutrients [14]. Therefore, pH is one major factor in the hydroponic system that needs to be monitored and controlled [15].

Hydroponically grown plants need different pH levels than plants grown in soil. Plants in hydroponic systems do not benefit from microorganism, organic matter and interaction between water and minerals which regulate pH. The hydroponic farmers must constantly monitor and adjust pH level and make sure not to apply pH for soil-grown plants to hydroponically grown plants. The number of optimal pH in the root zone of plants grown hydroponically ranges from 5.5 to 6.5 [16]. This is the range of pH where the nutrients are available to plants, although pH value as low as 4.0 is not prohibitive and considered to prevent the infections from Pythium and Phytophthora spp [17-20]. Generally, optimal pH range between 5.5 - 6.5 are for plants such as melons, apples, beans, squash and tomatoes. Meanwhile, blueberries need more acidic pH 4.0 - 5.0 [21, 22]. Too high and low pH create different plant responses. pH more than 6.5 and less than 7.5 (6.5<pH<7.5) affects the plant's ability to absorb nutrients and pH more than 7.5 (pH>7.5) decreases the availability of Iron, Manganese, Copper, Zinc and Boron [23]. Meanwhile low pH under 4.0 in the rhizosphere will inhibit plant growth due to high hydrogen (H+) activity directly. While the indirect impact of low pH level is the deficiency of nutrients such as phosphorus, molybdenum, calcium and magnesium which bring down the crop yields [24]. pH value should be consistent according to the need of the plant. Thus, the value of pH should be maintained in the optimum range according to each plant's need so that the plant grows faster and healthier than its natural growth [25]. However, it is suggested to separate nutrient reservoirs for the plant with similar pH range [26].

One of the systems used in hydroponics is DBS (Dutch Bucket System). DBS system is a hydroponic system used for plants with rooting mounts and suitable for plants that require nutrients in high amounts such as tomatoes. The working principle of the Dutch bucket is similar to the nutrient film technique (NFT) hydroponic system [27]. The DBS system uses planting containers with a large size and media used, namely hydroton and tile fragments [28]. The advantage of the DBS system is that the humidity of the plant root can be preserved well because the roots are always submerged in water stored in the container (bucket), yet DBS suffers from unbalanced nutrient acidity (pH) values [29]. The pH value of nutrients is always changing due to various factors such as planting media, bacteria, photosynthesis, and respiration processes. One method in the Dutch Bucket System is a recirculating system where a big reservoir holds the nutrient solution and is pumped through an irrigation line, then dropped onto the plants. The excess solution can return to the reservoir using the drain line [30, 31]. However, this recirculating system causes nutrient unbalance over time, so the nutrient should be replaced in the reservoir very often. This problem leads to the solution of controlling the acidity of the nutrients by the value of pH. Neglecting regular reservoir changes can conduct more drastic pH fluctuations resulting in limited availability of plant nutrients absorption [32]. So, in order to prevent pH fluctuations, the control of pH has come into attention.

For the past five years, there has been a rapid rise of solutions regarding pH control in hydroponic. IoT-based automation of controlling pH has been done on 40 green oak lettuce in greenhouse 1 x 2 meter using 12 fuzzy logic rules as the decision-making process to turn on or off four solutions used in DRFT (Dynamic Root Floating Technique) hydroponic system [29]. However, this system remains questionable for large scale implementation since it has no central processing. Central processing is important in automation for data processing to fetch data from memory or sensor, process them and store the results in memory and create a support for IoT nodes to perform pre-processing, discard useless information to perform complete features [33]. Besides, the system needs 80 - 90 minutes to achieve the desired pH value and not all of the fuzzy rules are working. Smart monitor and control system on hydroponic was done using Arduino Mega 328 to control the pH. It can accurately and transmit data of temperature, humidity, light intensity, water level and pH real time and the results are displayed in a graphical view [34]. However, there were no sensors used to control the nutritional needs of the plant although the monitoring can be seen through an application. Besides, the use of Arduino Mega 328 as the heart of the system shows inflexibility and unreliability of the system with many controlling parameters [35]. Reliability in

IoT is very crucial because any data loss or duplication can cause the IoT stack to fail which leads to life-threatening for end-users [36]. Fuzzy logic with Arduino UNO has been embedded to provide control on pH of the hydroponic system. The results show the significant growth of leaf width and plant height of lettuce and bok choy. This is also combined with ESP8266 for the Wi-Fi. Although this approach is interesting, this system lacks QoS analysis [37]. QoS (Quality of Service) parameter helps to know the capability of the network in transferring data. This will also help to improve the flaws in the network and provide better service to the users [38]. The shortcoming of internet connection dependence by using REST API protocol has been addressed in the nutrient solution acidity control with multiple linear regression methods. The system finds a deadlock when the internet connection is off [39]. The same problem of internet dependence also goes on [40] by using cloud computing and fog computing as the data processing in precision agriculture and farming in a rural area. Soil moisture monitoring and control system in precision agriculture using the Ubiquitous network has been conducted using the MQTT protocol in edge layer, while the REST API protocol for access to the cloud [41]. Although it seems good, it has only been implemented to control soil moisture only and yet to control pH.

In this context, we aim to provide edge-based and cloud computing acidity monitoring and control with the MQTT protocol QoS level 2 and sensors to help monitoring and controlling the nutrient such as sensor SHT-21, BH1750, JSN-SR04T, RTC DS3231, DS18B20, EC and pH Haosi H-101. We also undertook analysis on the work of the control system as well as QoS delay and jitter. Our focus is on pH value using pH Haosi H-101. Arduino Nano and NodeMCU ESP 8266 are used as the microcontroller. This system can work either with an internet connection (online) or without an internet connection (offline) by using Raspberry Pi as the central data, MQTT broker and gateway. We used tomatoes as the hydroponic plant which should range in pH of 5.5 - 6.5. Edge and cloud computing can help farmers monitor and control the acidity of nutrient solutions from a distance. The use of MQTT as a transmission protocol is for its light, open and simple characteristics, so it can be used in many situations including in the concept of Internet of Things (IoT) [42].

This paper is organized as follows: Section I is the introduction while Section II is system design. Result and discussion are explained in Section III and lastly, Section IV is the conclusion.

2.MQTT QoS Level 2

Our study uses MQTT QoS Level 2. The system is expected to successfully send and receive data so that the monitoring and control process runs well. Since, especially in the control system, reliability is needed so that the controller functions without fail, remaining functional and safe [43]. MQTT (Message Queuing Telemetry Transport) is one of the most used IoT communication protocols [44]. This brings message protocol based on publish/subscribe pattern. This pattern offers asynchronous communication in almost real-time. The publishers (clients) publish messages to a specific Topic connecting to a Broker (server) which forwards it to any subscribers of the topic [45]. Figure 1 shows the packet transmission of MQTT level. There are 3 levels of QoS in MQTT. QoS Level 2 is the highest MQTT service. QoS level 2 deliver the message through exactly once with 4-way handshake i.e. PUBLISH, PUBREC, PUBREL, PUBCOMP. By the 4-

way handshake mechanism, it is not possible to have message loss at this level. Thus, MQTT QoS Level 2 is considered the safest among QoS level 0 and 1. There is no guarantee of the message delivered in QoS level 0. Meanwhile in QoS level 1, there is a possibility of duplicate messages because of loss and no confirmation of the message being delivered [46].

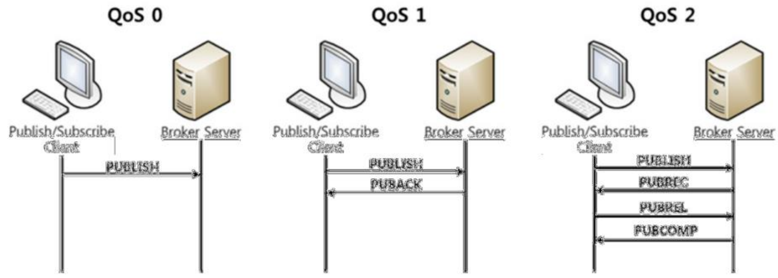

Fig. 1. MQTT QoS Level 0 - 2 [46].

3.Monitoring and Control System Design

Figure 2 shows the system architecture of monitoring and control. The monitoring system comprises sensors used as follows: SHT-21 sensor is used to measure the temperature and humidity of the greenhouse, BH1750 sensor is used to measure the greenhouse light intensity and JSN-SR04T sensor functions to measure the surface distance of the nutrient solution in the nutrient reservoir. RTC (Real Time Clock) helps to set the date and time in the system. NodeMCU reads the data from RTC, SHT-21, JSN-SR04T. Arduino IDE software was used to program the NodeMCU and Arduino Nano.

The monitoring program employed Raspberry Pi. The program on the Raspberry Pi was performed using Thonny Python IDE software to receive and store all the data in the local database and send the monitoring data to database server www.smart-gh.com. In this study, MQTT Broker Mosquitto was installed to activate MQTT Broker on the Raspberry Pi, while the Paho MQTT Client makes the Raspberry Pi function as a client with the python program that has been created. Next, MySQL, MariaDB, Apache and PHP are installed to function as databases on the Raspberry Pi (local database). The databases on the Raspberry Pi and on the www.smart-gh.com server are made the same to make synchronization easier.

The Python program begins by connecting to MQTT Broker and MySQL local database. The monitoring data collection process is carried out by subscribing to the topic "data". The data received by Raspberry Pi will then be sent to the local database and the www.smart-gh.com database server. If the Raspberry Pi is connected to the internet (online), the monitoring data will be sent to the Raspberry Pi local database MySQL and www.smart-gh.com database server. However, if it is not connected to the internet (offline), the monitoring data will only be sent to a local database.

The monitoring system is commenced by connecting NodeMCU to Access Point and MQTT broker. Once it is all connected, the reading of temperature and humidity using SHT21, light intensity BH1750 and distance of the nutrient reservoir with JSN-SR04T is started. Next, NodeMCU will send logic 0.1 to Arduino Nano to activate the relay for 10 seconds. The reading starts when Arduino Nano receives logic 1.0 that will light up the relay for 25 seconds to read temperature sensor DS218B20 and EC sensor. If the reading of sensors is successful, the next reading will be continued, while if there is no response from sensor reading, NodeMCU will wait for 60 seconds and proceed to the subsequent process i.e. reading pH value. NodeMCU sends logic 1.1 to Arduino Nano to read the pH sensor. If Arduino Nano receives logic 1.1., the reading of the pH sensor starts indicated by the relay lighting up for 35 seconds. Then, all the data read is sent to NodeMCU. NodeMCU will display all sensor measurements to a 20x4 I2C LCD screen and publish the topic “data” to MQTT broker. The data then is stored in the Raspberry Pi data center which can be accessed using Android on the edge side while offline. Once there is an internet connection (online), the last data stored in the Raspberry Pi data center will be sent to the data center server and displayed on the website interface.

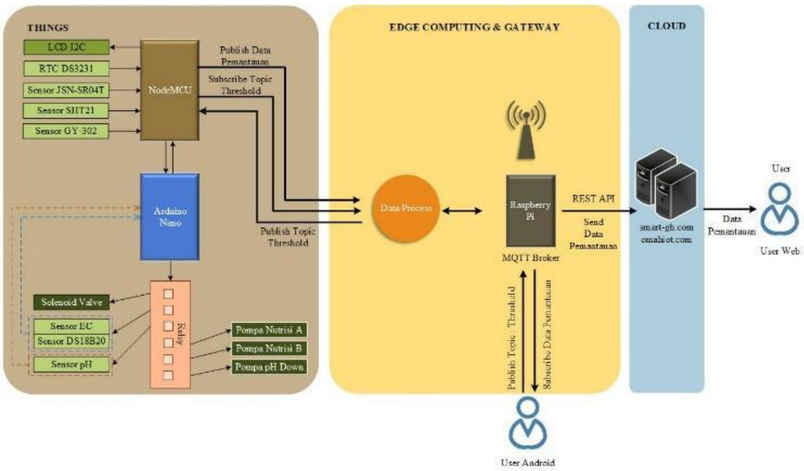

Fig. 2. Monitoring system architecture.

On the other hand, the control system works as a feedback response from the monitoring results of acidity nutrients by a pH sensor. Figure 3 shows the control system architecture.

The control process starts from the feedback of the monitoring system where the value of acidity nutrient measured by pH sensor was obtained. Then, according to Fig. 3., NodeMCU will retrieve the threshold data from Raspberry Pi by subscribing to the threshold topic through the broker. The threshold data contains the minimum threshold value for the acidity nutrient (pH) which is set by the farmer via Android according to the needs of the hydroponic grown plant. After the threshold data has successfully been retrieved, NodeMCU sends the data to Arduino Nano. Arduino Nano will compare the measured value of the pH sensor with the received threshold value. If the measured value of the pH sensor exceeds the threshold value, the control system will work by activating a relay connected to a DC pump actuator. The pump will drain the pH down solution into the main reservoir through a hose.

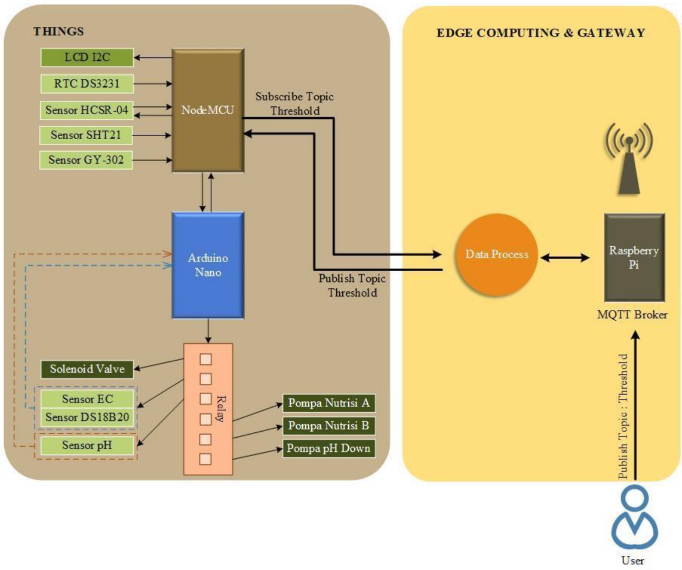

Fig. 3. Control system architecture.

Before the testing was performed, all the sensors used have been calibrated and they can work well. The pH Haosi H-101 has an accuracy of 98.92%, the SHT-21 sensor has an accuracy of 84.03% humidity and a temperature accuracy of 96.09%, the BH1750 sensor has an accuracy of 95.76%, the JSN-SR04T sensor has an accuracy of 95.76%. of 97.08% and the DS18B20 sensor has an accuracy of 98.43%. These results indicate that the sensors work well and feasible to use.

4.Testing of Control System

The control system testing was carried out with 20 times of data retrieval at each threshold that has been determined by the user via Android. The threshold values used are 5.7, 6, 6.3. Tests were carried out to determine whether the control system will turn on when the pH value exceeds a predetermined threshold and will turn off if it is less than/equal to the threshold value. The pH start value is the pH value before controlling and the pH end value is the value after control.

Testing of Quality of Service (QoS)

QoS testing was carried out to determine the network quality of the system that has been made. The testing of QoS includes delay and jitter. This test was done 24 hours to understand the quality of the system network created. The delay test was carried out to find out how long it takes to send monitoring data from NodeMCU to Raspberry pi (local database) and from NodeMCU to cloud (database server). The delay value is obtained by calculation based on Equation 1.

$$Delay = (receive\ time - send\ time) seconds \tag{1}$$

The jitter testing was carried out to determine the delay variation or the variation in the arrival time of the packet. The jitter value is obtained by calculation based on Equation 2 and the total variation of delay is calculated using Equation 3.

$$Delay = \frac{total\ delay\ variation}{total\ data\ received - 1} \quad (2)$$

$$\sum delay\ variation = (delay\ 2 - delay\ 1) + \cdots + (delay\ n - delay\ (n-1)) \quad (3)$$

5. Results and Discussion

5.1. Control system performance

The threshold of pH that has been set for the testing are 5.7, 6, and 6.3. Figure 4 shows the results of control system performance which compare the threshold of pH and pH end value.

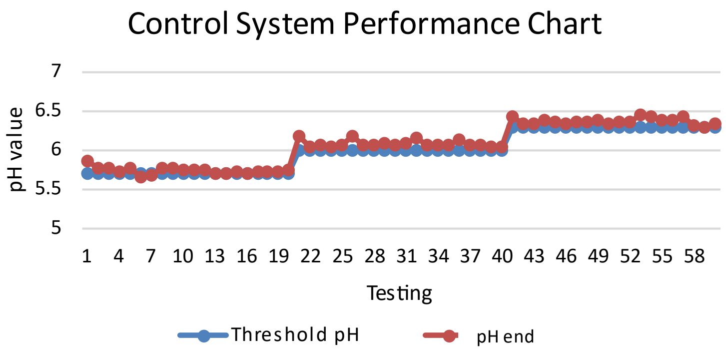

Fig. 4. Control system architecture.

Figure 4 is controlled system performance. To understand whether there is a difference on the value of pH with threshold 5.7, 6, and 6.3 in its pH end (after control system applied), we conducted a normality test first to know whether the data are normally distributed or not shown by p-value [47]. If p-value is less than 0.05, meaning that the data is not normally distributed, thus the analysis is changed to non-parametric Wilcoxon test. Otherwise, if the p-value is more than 0.05, the analysis uses simple T-Test. Table 1 shows the normality test of the pH using software SPSS 25. Because the sample is less than 30, then, in the normality test, we evaluate the p-value in the Shapiro-Wilk column.

The results of Table 1 show the p-value of more than 0.05 in threshold 5.7 and threshold 6.3 and less than 0.05 in threshold 6. If there is data with p-value of more than 0.05 and less than 0.05, meaning that the data are not normally distributed. Therefore, the Wilcoxon test was chosen to analyse whether there is a significant difference or not on each threshold and pH end to conclude that the control system has influence in the change of pH value.

Table 1. Normality test results on threshold and pH end.

	Kolmogorov-Smirnov[a]			Shapiro-Wilk		
	Statistic	**df**	**Sig.**	**Statistic**	**df**	**Sig.**
Thd 5.7_after	.128	20	.200*	.935	20	.196
Thd 6_after	.278	20	.000	.793	20	.001
Thd 6.3_after	.145	20	.200*	.966	20	.666

*. This is a lower bound of the true significance.

a. Lilliefors Significance Correction

Table 2 shows the results of Wilcoxon test on threshold and pH end.

Table 2. Wilcoxon test results on threshold and pH end.

Test Statistics[a]			
	Thd_5.7_after - Thd_5.7	**Thd_6 after - Thd 6**	**Thd_6.3 after - Thd 6.3**
Z	-3.209[b]	-3.935[b]	-3.829[b]
Asymp. Sig. (2-tailed)	.001	.000	.000

a. Wilcoxon Signed Ranks Test
b. Based on negative ranks.

Based on Table 2, it can be seen that all the threshold set value and the pH end show p-value of less than 0.05. Meaning that, there is a significant difference between the pH end and its threshold. So, it means that the control system has influence in the change of pH value. Thus, the control system works well. This is also supported with Table 3 which shows the accuracy of the control system. The control system provides the average accuracy of 98.84% among all thresholds: 5.7, 6 and 6.3.

Table 3. Accuracy of the control system.

Threshold	Error (%)	Accuracy (%)
5.7	0.81	99.19
6	1.45	98.55
6.3	1.23	98.77
Average	**1.16**	**98.84**

5.2. Quality of service (QoS) delay

The delay test is carried out to find out how long the travel time and variations in the process of sending monitoring data are from NodeMCU to Raspberry pi (local database) and from NodeMCU to cloud (server database). This test was done to test the system thoroughly (after the control and monitoring system were put together).

Table 4. QoS delay.

Time	Edge Average *Delay* (seconds)	Cloud Average *Delay* (seconds)
00.00.01 - 06.00.00	1.49	1.76
06.00.01 - 12.00.00	1.64	2.05
12.00.01 - 18.00.00	1.67	2.52
18.00.01 - 00.00.00	1.68	4.79
Total Average	**1.62**	**2.78**

According to Table 4, the average delay on the edge is 1.62 seconds and on the cloud, the average value of the delay is 2.78 seconds. Based on the delay category, the average delay either on the edge and cloud are categorized as bad in index 1, because the delay is more than 450 ms.

The bad delay in this control system is caused by the implementation of MQTT QoS level 2. Since, in the level 2, there are 4-way handshake of data exchange including the process of PUBLISH, PUBREC, PUBREL, PUBCOMP. The advantage of this level 2 is that there is no possibility to have a message loss. However, due to the complicated process of 4-way handshake, relatively longer end-to-end delays are possible [46]. This makes the delay value in our study categorized as bad by as much as 1.62 seconds in edge and 2.78 seconds in cloud. It is the compensation of no message loss. Besides, the use of wireless networks may cause longer delays than wired networks. This is in line with the research done by [46] evaluating the correlation between end-to-end delay in relation to QoS level 2. In wired networks, QoS 2 shows a significant change in end-to-end delay with messages smaller than 8,000 bytes. Nevertheless, the research confirms that by using QoS level 2, the message loss was reduced 1.57 times than by using QoS level 0 or 1. Meanwhile in the wireless network, the end-to-end delay shows longer delays. The contributing factor of the delay is also determined by the transmission speed of the network, in this case 3G network. Wireless network is confirmed to result in longer delay compared to wired network due to the 3G network connection. Both wired and wireless networks show the same results that QoS level 2 provides a lower message loss rate than QoS level 0 or 1.

In the automation era, the use of wireless has been widespread, such as the use of 802.11 technology (WiFi) and Bluetooth (802.15.1), Wireless Local Area Network was used in our study, and thus the standard we used is 802.11. The use of WLAN includes the use of TCP (Transmission Control Protocol) support. TCP provides more reliability in MQTT, since MQTT is designed for reliability and not for speed. Instead, MQTT has a strong advantage of scalability. It supports a large number of small, constrained devices and a way to ensure asynchronous communication of devices. An analysis of the lowest possible packet delay for 802.11 standard is 0.2 ms and for 802.15.1 is 1.25 ms. This is important information for automation control applications, where we can refer to the results to determine which of the technology we will use in the control system, where in our study the use of WLAN in the control system provides average delay of 1.62 seconds and 2.78 seconds with MQTT protocol.

However, the use of wireless networks is truly a big challenge to transmit data over wireless networks due to the limited channel capacity, unstable channel condition as well as interference or congestion. This is the function of edge computing. The proximity of the edge itself allows data processing to and from remote clouds to be handled at the edge. So by the use of edge or processing data locally through an accelerated data stream will possibly reduce the network traffic bottleneck. Table 4 indicates the average delay on the edge is lower than in cloud. This is because on the edge side of data processing is done at the edge of the network so that it can be processed directly as edge is a middle layer between end user and cloud. Meanwhile, on the cloud side, data processing must go through an internet provider so that the data transmission process takes longer.

5.3. Quality of service jitter

Jitter evaluation is done also in 24 hours at the same time of delay evaluation. Table 5 shows the QoS jitter of the system. It indicates that the edge average jitter is 0.49 seconds and in the cloud, the average jitter shows 1.64 seconds.

The values in Table 5 show that the average jitter both in edge and cloud show different values where the edge average jitter is less than that in the cloud. According to the category of jitter values of ETSI (1999), edge average jitter of

0.49 seconds and cloud average jitter with 1.64 seconds are categorized as bad since the value is over the bad category of 125 - 225 ms with index 1. The causes of the high value of jitter include a sudden increase in traffic and the magnitude of collisions between packets (congestion). The reason for average jitter in edge obtain less than in cloud is the use of edge computing which accelerates the data processing before going to the cloud computing.

Table 5. QoS jitter.

Time	**Edge Average *Jitter* (seconds)**	**Cloud Average *Jitter* (seconds)**
00.00.01 - 06.00.00	0.40	0.61
06.00.01 - 12.00.00	0.70	0.92
12.00.01 - 18.00.00	0.58	1.08
18.00.01 - 00.00.00	0.31	3.93
Total Average	**0.49**	**1.64**

Delay, jitter, throughput, packet loss ratio and fairness are known as popular parameters. However, delay and jitter are the most important parameters of Quality of Service (QoS) since they have significant impact on network QoS. We can observe that most of the value of delay is correlated to the value of jitter in which the higher the delay, the higher the jitter. The reason behind this condition is that delay and jitter are innately linked. To prove that delay and jitter are correlated, we conducted a correlation test using SPSS 25 with the input of delay and jitter values in Table 4 and 5.

Table 6 shows the correlation between delay and jitter on QoS. According to the Sig.2(value), if the value is more than 0.05, there is no correlation, otherwise, if the value is less than 0.05, then there is a correlation. The Sig.2(value) of delay and jitter obtain 0.000 which is less than 0.05, thus there is a significant correlation between delay and jitter. The same result is obtained between jitter and delay which show Sig.(2-tailed) value of 0.000. Therefore, it is true that there is correlation between jitter and delay. Meanwhile from the value of Pearson correlation, the value of delay and jitter shows correlation value 0.988. This value shows the strong correlation between delay and jitter because the value is more than 0.5. The same strong correlation is shown between jitter and delay with the Pearson correlation value of 0.988. The positive values of Pearson correlation indicate that most of the higher the delay value, the higher the jitter value. Even though delay and jitter are correlated, they are not the same in definition. Delay is time for data to move from one endpoint on the network to another. Meanwhile jitter is the difference in delay between two packets.

Table 6. Correlation statistical tests between delay and jitter.

correlations.

		Delay QoS	**Jitter QoS**
Delay QoS	Pearson Correlation	1	.988**
	Sig. (2-tailed)		.000
	N	8	8
Jitter QoS	Pearson Correlation	.988**	1
	Sig. (2-tailed)	.000	
	N	8	8

**. Correlation is significant at the 0.01 level (2-tailed).

6. Conclusions

Based on the research result, some conclusions can be seen below:

- Automation will help farmers to grow hydroponic plants since there is no need to check the plant manually all the time. We have made edge and cloud computing monitoring and control system of pH value in Dutch Bucket System hydroponic using MQTT protocol QoS level 2 by using microcontroller of NodeMCU ESP 8266 and Arduino Nano and the pH sensor of Haosi H-101.
- The system can function properly according to control pH value in three thresholds i.e. 5.7, 6 and 6.3 proven by Wilcoxon statistical test resulting in p-value less than 0.05 and with an average control system accuracy of 98.84% based on the difference of pH start and pH end. However, the Quality of Service (QoS) value in delay and jitter are categorized as bad with average value of delay 1.62 seconds in edge computing and 2.78 seconds in cloud computing. Meanwhile the jitter obtains 0.49 in edge computing and 1.64 in cloud computing.
- We have found that the bad category of delay and jitter are the consequence of MQTT protocol QoS level 2 due to 4-way handshake to guarantee the message delivered since MQTT is designed for reliability not for the speed to support scalability in large number of small, constrained devices and providing a simple way for asynchronous communication of devices.

References

1. Ditta, A.; Arshad, M.; and Ibrahim, M. (2015). Nanoparticles in sustainable agricultural crop production: Applications and perspectives *Nanotechnology Plant Science*, 55-75.
2. Selim, M.M. (2020). Introduction to the integrated nutrient management strategies and their contribution to yield and soil properties. *International Journal of Agronomy*, Volume 2020, Article ID 2821678, 1-14.
3. Statistik, B.P. (2019). *Direktori Perusahaan Pertanian Peternakan 2019*. Jakarta: Badan Pusat Statistik Indonesia.
4. Goulding, K.; Jarvis, S.; Whitmore, A.; and Ex, D. (2008). Optimizing nutrient management for farm systems. *Philosophical Transactions of the Royal Society B*, 2007, 667-680.
5. Tuğrul, K.M. (2019). Soil Management in Sustainable Agriculture. *IntechOpen*, 1-16.
6. Sharpley, A. (2016). Managing agricultural phosphorus to minimize water quality impacts. *Scientia Agricola*, 73(1), 1-8.
7. Nursyahid, A.; Aprilian, T.; Setyawan, T.A.; Helmy; Nugroho, A.S.; and Susilo, D. (2019). Automatic sprinkler system for water efficiency based on LoRa network. *Proceedings of the 2019 6th International Conference on Information Technology, Computer and Electrical Engineering (ICITACEE)*. Semarang, Indonesia, 1-6.
8. Crisnapati, P.N.; Wardana, I.N.K.; Aryanto, I.K.A.A.; and Hermawan, A. (2017). Hommons: Hydroponic management and monitoring system for an IOT based NFT farm using web technology. *Proceedings of the 2017 5th*

International Conference on Cyber and IT Service Management (CITSM). Denpasar, Indonesia, 1-6.

9. Sudana, D.; Eman, D.; and Suyoto. (2019). IOT based: Hydroponic using drip non-circulation system for paprika. *Proceedings of the 2019 International Conference of Artificial Intelligence and Information Technology (ICAIIT)*. Yogyakarta, Indonesia, 124-128.
10. Langelier, W.F. (1946). Effect of temperature on the pH of natural waters. *Journal of American Water Works Association*, 38(2), 179-185.
11. Guo J.H.; Liu, X.J.; Zhang, Y.; Shen, J.L.; Han, W.X.; Zhang, W.F.; Christie, P.; Goulding, K.W.T.; Vitousek, P.M.; and Zhang, F.S. (2010). Significant acidification in major Chinese croplands. *Science,* 327(5968), 1008-1010.
12. Anderson, T.S.; Martini, M.R.; De Villiers, D.; and Timmons, M.B. (2017). Growth and tissue elemental composition response of butterhead lettuce (Lactuca sativa, cv. Flandria) to hydroponic conditions at different pH and alkalinity. *Horticulturae*, 3(41), 1-15.
13. Omer, N.H. (2019). Water quality parameters. *IntechOpen,* 1-19.
14. Neina, D. (2019). The role of soil pH in plant nutrition and soil remediation. *Applied Environmental Soil Science*, Volume 2019, Article ID 5794869, 1-9.
15. Maldonado, A.I.L.; Reyes, J.M.M.; Breceda, H.F.; Fuentes, H.R.; Contreras, J.A.; and Vidales, U.L.M. (2019). Automation and robotics used in hydroponic systems. *IntechOpen*, 1-24.
16. Alexopoulos, A.A.; Marandos, F.; Assimakopoulou, A.; Vidalis, N.; Petropoulos, S.A.; and Karapanos, I.C. (2021). Effect of nutrient solution pH on the growth, yield and quality of taraxacum officinale and Reichardia picroides in a floating hydroponic system. *Agronomy,* 11(6), 1118, 1-15.
17. Gillespie, D.P.; Kubota, C.; and Miller, S.A. (2020). Effects of low pH of hydroponic nutrient solution on plant growth, nutrient uptake, and root rot disease incidence of basil (Ocimum basilicum L.). *HortScience*, 55(8), 1251-1258.
18. Singh, H.; Dunn, B.; Payton, M.; and Brandenberger, L. (2019). Fertilizer and cultivar selection of lettuce, basil, and Swiss chard for hydroponic production. *Horttechnology*, 29(1), 50-56.
19. Arora, H.; Sharma, A.; Sharma, S.; Haron, F.F.; Gafur, A.; Sayyed, R.Z.; and Datta, R. (2021). *Pythium* damping-off and root rot of *Capsicum annuum L.*: Impacts, diagnosis, and management. *Microorganisms*, 9(4), 823, 1-17.
20. Rajatha, K.D.; Prasad, S.R.; Kumar, M.U.; and Nethra, N.M.N. (2019). Standardization of pH of the solution under soil-less agriculture system for tomato (Solanum lycopersicum L.). *International Journal of Current Microbiology and Applied Science*, 8(12), 1111-1118.
21. Jones, J.J.B. (2004). *Hydroponics: A practical guide for the soilless grower*. Florida: CRC Press.
22. Singh, H.; and Bruce, D. (2016). Electrical conductivity and pH guide for hydroponics. *Oklahoma Cooperative Extension Service*, 1-5.
23. Lenord Melvix. J.S.M and Sridevi, C. (2014). Design of efficient hydroponic nutrient solution control system using soft computing based solution grading. *Proceedings of the 2014 International Conference on Computation of Power, Energy, Information and Communication (ICCPEIC).* Chennai, India, 148-154.

24. Zhang, Y.; Zhu, D.; Zhang, Y.; Chen, H.; Xiang, J.; and Lin, X. (2015). Low pH-induced changes of antioxidant enzyme and ATPase activities in the roots of rice (Oryza sativa L.) seedlings. *PLoS One*, 10(2), 1-12.
25. Palande, V.; Zaheer, A.; and George, K. (2018). ScienceDirect fully automated hydroponic system for indoor plant growth. *Procedia Computer Science*, 129, 482-488.
26. Yang, T.; Samarakoon, U.; Altland, J.; and Ling, P. (2021). Flavor-related phytochemical properties of hydroponic-grown arugula (Eruca sativa Mill.) 'Standard' under different electrical conductivities of nutrient solution. *Agronomy*, 11(7), 1340, 1-20.
27. Helmy; Mahaidayu, M.G.; Nursyahid, A.; Setyawan, T.A.; and Hasan, A. (2017). Nutrient film technique (NFT) hydroponic monitoring system based on wireless sensor network. *Proceedings of the 2017 IEEE International Conference on Communication, Networks and Satellite (Comnetsat)*. Semarang, Indonesia, 81-84.
28. Macwan, J.; Pandya, D.; Pandya, D.H.; and Mankad, D.A. (2020). Review on soilless method of cultivation: hydroponics. *International Journal of Recent Scientific Research*, 11(01), 37122-37127.
29. Fuangthong, M.; Pramokchon, P. (2018). Automatic control of electrical conductivity and pH using fuzzy logic for hydroponics system. *Proceedings of the 2018 International Conference on Digital Arts, Media and Technology (ICDAMT)*. Phayo, Thailand, 65-70.
30. Rasheed, J.; Latif, K.; Sheraz, M.; and Iqbal, A. (2021). Building indigenous smart hydroponic farm as lessons from an academic experiment--A review article. *International Journal of Agricultural Technology*, 17(2), 673-684.
31. Sace C.F.; and Estigoy, J.H. (2015). Lettuce Production in a Recirculating Hydroponic System. *American Journal of Agricultural Science*, 2(5), 196-202.
32. Toyosumi, S.; Santos, T.; Melo, D.M.; Dias, A.; Neto, D.A.; and Miler, T. (2021). Optimization of banana plantlets acclimatization by hydroponic cultivation. *Scientia Agricola*, 78(e20200165), 1-9.
33. Capra, M.; Peloso, R.; Masera, G.; Roch, M.R.; and Martina, M. (2019). Edge computing: A Survey on the hardware requirements in the internet of things world. *Future Internet*, 11(100), 1-25.
34. Charumathi, S.; Kaviya, R.M.; Kumariyarasi, J.; Manisha, R.; and Dhivya, P. (2017). Optimization and control of hydroponics agriculture using IOT. *Asian Journal of Applied Science and Technology (AJAST)*, 1(2), 96-98.
35. Nalwade, R.; and Mote, T. (2017). Hydroponics farming. *Proceedings of the 2017 International Conference on Trends in Electronics and Informatics (ICEI)*. Tirunelveli, India, 645-650.
36. Moore, S.J.; Nugent, C.D.; Zhang, S.; and Cleland, I. (2020). IoT reliability: a review leading to 5 key research directions. *CCF Transactions on Pervasive Computing and Interaction*, 2(3), 147-163.
37. Herman; and Surantha, N. (2020). Smart hydroculture control system based on IoT and fuzzy logic. *International Journal of Innovative Computing, Information and Control*, 16(1), 207-221.
38. Sheikh, A.; Ambhaikar, A.; and Kumar, S. (2021). Analysis of QoS parameters for IoT networks. *Open Journal of Science and Technology (OJST)*, 4(1), 40-50.

39. Helmy, H.; Janah, D.A.M.; Nursyahid, A.; Mara, M.N.; Setyawan, T.A.; and Nugroho, A.S. (2020). Nutrient solution acidity control system on NFT-based hydroponic plants using multiple linear regression method. *Proceedings of the 2020 7th International Conference on Information Technology, Computer, and Electrical Engineering (ICITACEE)*. Semarang, Indonesia, 272-276.
40. Ahmed, N.; De, D.; Member, S.; and Hussain, I. (2018). Internet of things (IoT) for smart precision agriculture and farming in rural areas. *IEEE Internet Things Journal*, 5(6), 4890-4899.
41. Ferrández-Pastor, F.J.; García-Chamizo, J.M.; Nieto-Hidalgo, M.; Mora-Pascual, J.; and Mora-Martínez, J. (2016). Developing ubiquitous sensor network platform using internet of things: application in precision agriculture. *Sensors*, 16(7), 1141.
42. Mishra, B.; and Kertesz, A. (2020). The use of MQTT in M2M and IoT systems : A survey. *IEEE Access*, 8, 201071-201086.
43. Nguyen, V.-A.-O.; and Yoo, M. (2015). Packet loss compensation for control systems over industrial wireless sensor networks. *International Journal of Distributed Sensor Networks*, 11(8), 1-9.
44. Battistoni, P.; Sebillo, M.; and Vitiello, G. (2021). IoT An IoT-based mobile system for safety monitoring of lone workers. *IoT*, 2(3), 476-497.
45. Light, R.A. (2017). Mosquitto: server and client implementation of the MQTT protocol. *The Journal of Open Source Software*, 2(13), 1-2.
46. Lee, S.; Kim, H.; Hong, D.; and Ju, H. (2013). Correlation analysis of MQTT loss and delay according to QoS level. *Proceedings of the International Conference on Information Networking 2013 (ICOIN)*. Bangkok, Thailand, 714-717.
47. Ghasemi A.; and Zahediasl, S. (2012). Normality tests for statistical analysis: A guide for non-statisticians. *International Journal Endocrinology Metabolism*, 10(2), 486-489.