Industrial Internet of Things - Control of Industry Simulation Through Different Protocols

P Surya1, Asuman Suenbuel2, Arockia Selvakumar Arockia Doss4,*

1School of Mechanical Engineering, Vellore Institute of Technology, Chennai, India
2SAP labs, USA

*arockia.selvakumar@vit.ac.in

Abstract. In past few years there is a great demand for internet of things (IoT), It has become an important part of smart technologies. There have been many researches going-on in internet of things in both academics and industries. IoT using in industries is also known as Industrial Internet of Things (IIoT). Where smart sensors are used along with IoT are used. This research work is done to identify the best protocol for high-speed data transmission with no loss of data to prevent command lag between mobile controller and industry. In this research a soda filling Industry simulation with 3 different plants was controlled through a mobile application through different protocols with 4 different features of data transfer. The time of data sending and receiving from application and ESP32 controller are stored in a log. The speed of data transfer of all different protocols is compared. From the final result, ThingSpeak is an optimum protocol for this application which is 36% faster than HTTP in single data transfer and 39% faster than MQTT in continuous data transfer.

1. Introduction

Industrial Internet of Things is a part of industrial revolution 4.0 [1][12], in which all the machine are interconnected (M2M), and the process planning and inventory are controlled automatically by the central computer [2][3], all these are connected through internet with high secure data transfer [4]. In this paper a soda bottle filling industry model simulation is made in Flexsim and interface ESP32 with Flexsim through Python API by serial port and it is controlled through a mobile application as shown in figure 1. The forecasting model and prediction model are made to test the data transfer [5][6]. The data for these models are stored in different space. The communication between the application and ESP32 are made through different protocols and compared.
2. Industry Simulation

The Simulation model of an industry is modelled and simulated in Flexsim simulation software [7][8] as in figure 2. Soda bottle filling industry was simulated in this research work. The soda bottle filling station contained three plants in it.

2.1. Bottle Collection Plant

In Bottle Collection Plant, the used bottle from the users will be collected by the dealers will be received by truck and cleaned and check for label and send to filling plant process is shown in figure 3.
2.2. Bottle Producing Plant
The Bottle Producing plant produce empty bottles from the glass scarps. This plant takes input the scrap received by a truck is melted in furnace and using blow casting technique the bottles will be produced and the bottle will be cooled and labelled and Stored in the Inventory as pallets as shown in figure 4.

Figure 4. Bottle Producing Plant Process Flow.

2.3. Bottle Filling Plant
The Bottle Filling Plant, takes input the empty bottles and outputs the filled bottles. The plant takes the bottle from both Bottle collection plant and Bottle producing plant to match the production process. Then on next station, the collected bottles will be then cleaned and then on next station filling and capping of the bottle will be done after that the bottle level will be inspected in the next machine, improper filled bottles will be rejected in this process and then the bottles are stacked in the pallets in pelleting station and it will be stored in the inventory is shown in figure 5. The final product in the inventory will be send to dealers through trucks. Flexsim simulation model is shown in Figure 6.

Figure 5. Bottle Filling Plant Process Flow.

Figure 6. Flexsim Simulation Model of the Bottle Filling Industry.
3. Mobile Application

The Mobile Application was designed and programmed in MIT App Inventor using drag and drop interfaces. First the mobile Application was designed with the required number of options and tabs shown in figure 7. Then the application was programmed to send data through cloud to the ESP32 controller. The ESP32 receives the data and send it to the simulation model through serial port. The mobile application and ESP32 are also programmed to store the time at which the value is send and received in a log file. Total of three applications are designed with same design but different protocol communication programming.

![Figure 7. Mobile Application Developed for displaying Simulation Result.](image)

The ESP32 module is a Bluetooth and Wi-Fi Microcontroller Module, ESP32 is used in real-time data transferring from machine to cloud. Since this paper the machine itself a simulation, the data transfer is from simulation to ESP32 by Serial communication and ESP32 to mobile application through cloud. And similar to mobile application. ESP32 also programmed for three separate protocols.

4. Protocols

IoT Communication Protocol is mode of communication used for data transferring in a secure manner between devices. The IoT devices are mainly connected to internet via Internet Protocol (IP) network. IoT devices also work in non-IP networks which includes Bluetooth and RFID, Non-IP network communication are mainly used in Machine-to-Machine communication for short range data transfer. Non-IP network also consume very less power and simple to program. IP network uses internet as its base. IP network communication required lot power than non-IP but the range of data transfer is limitless through internet. In this work three IP Protocols HTTP and MQTT and combination of both are used.
4.1 HTTP (Hyper Text Transfer Protocol)
Hypertext Transfer Protocol act over the web as its base for data communication. HTTP protocols act as a communication platform between Client and server [9][14]. Client which transfers a data form a URL with the data and server which receives it use this URL to receive the data and send a response message to the client. HTTP protocol method was used to program both Mobile Application and ESP32 module and data are collected.

4.2 MQTT (Message Queuing Telemetry Transport)
Message Queuing Telemetry Transport is also one of the IP networks, this protocol has a main MQTT broker which store the data and clients which sent and receive data through MQTT Broker [10][13][16]. The data stored can be received by any number of clients. ESP32 and Mobile application are programmed using MQTT Protocol and data are collected.

4.3 ThingSpeak (MQTT and HTTP)
ThingSpeak Cloud uses both MQTT and HTTP protocol. It uses MQTT broken in MQTT protocol for storing data but uses HTTP protocol for communication. Using ThingSpeak also the ESP32 and Mobile Application are programmed and data are collected [11][15].

5. Features in Mobile Application

5.1 Live Output of Industry
The Home page of the Application displays the Live output of industry i.e., the product outputted from each machine and number of products accepted and number of products rejected. In this time at which data send from ESP32 is stored and data received by the mobile application is also stored.

5.2 Demand Forecasting Model
The Demand forecasting model is used to forecast the demand of the company on the upcoming year, by forecasting the demand was used to plan and update the machine or install a new assembly line the industry to match the demand to be produced. In this application Vector Autoregression Moving-Average with Exogenous Regressors (VARMAX) algorithm was used for forecasting the demand. This algorithm is programmed in Python environment and the forecasted demand will be uploaded to the Cloud. The mobile will download the forecasted demand from cloud and display in the Forecast Tab in Application.

5.3 Maintenance Schedule
The machine used in the industries need a Maintained regularly to prevent breakdown in the production line. There are many maintenance methods used in the industries, some are Run-to-failure, Preventive, Predictive and Reliability-centred maintenance. In this paper, preventive maintenance was used for the simulation, Preventive maintenance are maintenance provided to the machine in scheduled manner. The machine component life cycle that is provided by the supplier and also the data collected from experience persons will be collected and the collected data will be converted to products per life and stored in the server as database.

During the production process the maintenance model monitor the product produced and the product produced will be subtracted from the life of the component. If the component is too low then the program checks the life of other components as well, then form a group for components to be replaced, since production can’t be stopped for each component so that maintenance was scheduled group of components such that unwanted frequent maintenance can be prevented. The industry has 3 production line and two active lines and one for backup, when one of the lines is in maintenance then other two line will act as active lines. When maintenance is done the new life data will updated to the database.
5.4 Predictive Model

The predictive model is used to predict the time taken for producing certain amount of product in the industry. The predictive model is programmed in python, this program takes the product to be produced as input and it consider the product produced in the line. Then outputs the time required to produce the given input of products as number of days.

By using each feature, the data sending and receiving time will be stored in a log file with a data ID for each communication. The data can be either input to the model or output from the model. Live Output of industry shows the transfer of continuous data only with an interval of 5 secs between data. Whether as the other features have long interval between each data.

6. Results and Discussion

The soda bottle filling industry simulation was modelled in Flexsim software by cloning the real-time industry. A mobile application was also developed for controlling this simulation model. Communication was formed between the mobile application and the simulation by ESP32 controller and python program.

Three different protocols are selected and used for this cloud communication and time at which the data was send and received with data ID are stored in a log file in both ESP32 and mobile application. The log files of each protocol are collected and the time difference (Received Time – Send Time) for each data ID is calculated. Similar process was used for all the features in the model, the results obtained are compared and shown below. The result shown is the Time vs Data ID graph for each feature in the model. Figure 8 shows the communication speed comparison of protocols for the forecasting feature. Forecasting feature is a one-way single data transfer i.e., the communication is only from simulation to the mobile application through cloud and time interval between data is more than 5 seconds.

**Figure 8.** Speed Comparison between Protocols for Single Data Transfer - One way Communication.

**Figure 9.** Speed Comparison between Protocols for Single Data Transfer - Two-way Communication.
Figure 10. Speed Comparison between Protocols for Continuous Data Transfer – One way Communication.

Figure 11. Speed Comparison between Protocols for Continuous Data Transfer – Two-way Communication.

Figure 9 shows the speed of protocols for predictive model, a two-way single data transfer, in this data transfer is done by both mobile application and simulation at a time and time interval between data is more than 5 seconds. Similarly figure 10 and figure 11 shows the speed comparison of protocols for Live output (One-way continuous data transfer) and Maintenance model (Two-way continuous data transfer) respectively. In continuous data transfer, the time interval between data are not more than 5 seconds. By the result obtained, the average and deviation of each protocol in all features are calculated and compared to find the best protocol for this application.

7. Conclusion

The research of Controlling simulation with different protocol by Industrial IoT was proposed and the following are concluded by the results obtained. The MQTT protocol is 44% faster than HTTP for Single Data transferring in One-way communication and also in Two-way communication and HTTP protocol is 41% faster in multi-Data or continuous data transferring in both one-way and Two-way communication. ThingSpeak is 36% faster than HTTP in single data transferring which is almost as good as MQTT and also 39% faster than MQTT, as good as HTTP in Continuous data transferring. So, ThingSpeak cloud (HTTP and MQTT) is optimum protocol for this application.

References

[1] Abdul-Qawy and Pramod P J 2015 The Internet of Things (IoT): An Overview (Hyderabad: Kakatiya University) Vol. 5 pp 71-82
[2] Mazhar R and Awais A 2016 IoT-based smart city development using big data analytical approach (Korea: Kyungpook National University)
[3] Kevin L and Khaled T 2016 A low-power IoT framework: From sensors to the cloud (USA: Central Michigan University)
[4] Feng Z and Min L 2016 An IoT-Based Online Monitoring System for Continuous Steel Casting (China: Tongji University)
[5] Temesegen W A and Rutvik M 2018 Air pollution monitoring and prediction using IoT (Vadodara: Parul University)
[6] Suresh M and Saravana Kumar P 2015 IoT Based Airport Parking System (Tamil Nadu: Bannari Amman Institute of Technology)
[7] Santhosh K and Mahesh 2015 Modelling and Analysis of Flexible Manufacturing System with FlexSim (Warangal: S.R Engineering College)
[8] Zhu X and Zhang R 2014 *A Flexsim-based Optimization for the Operation Process of Cold-Chain Logistics Distribution Centre* (China: Beijing Jiaotong University)

[9] Mah B A 1997 *An empirical model of HTTP network traffic* (Berkeley: University of California at Berkeley)

[10] Roger A L 2017 *Mosquito: server and client implementation of the MQTT protocol*

[11] Sharmad P 2016 *Thingspeak Based Sensing and Monitoring System for IoT with MATLAB Analysis* (Bengaluru: College of Engineering and Technology) Vol. 2 pp 19-23

[12] Jalamkar, D., and AA Selvakumar. "Use of internet of things in a humanoid robot-a review." *Advances in Robotics & Automation* 5, no. 2 (2016): 2-5.

[13] Sengodan, Prabaharan, and Yousef Hasan F. Jbara. "Development of IoT Controlled Agri-Rover for Automatic Seeding." *International Journal of Pure and Applied Mathematics* 114, no. 11 (2017): 241-251.

[14] Suryavamsi, P. S. N., and A. Arockia Selvakumar. "IoT controlled mobile robot for home security and surveillance." In *International Conference on Intelligent Data Communication Technologies and Internet of Things*, pp. 431-438. Springer, Cham, 2018.

[15] Jalamkar, D., and Arockia Selvakumar A. Krishnakumar. "Implementation of Internet of things in a mobile Humanoid robot: A base for future of IOT enabled robotics applications." *International Journal of Engineering & Technology* 7, no. 4.36 (2018): 386-389.

[16] Palit, Swastika, Shreya Sinha, R. Aylmer Britto, and A. Arockia Selvakumar. "CFD Analysis of flow around fish." In *Journal of Physics: Conference Series*, vol. 1276, no. 1, p. 012010. IOP Publishing, 2019.