2019 IEEE Student Conference on Research and Development (SCOReD)
October 15-17, 2019, Seri Iskandar, Perak, Malaysia

Remote Monitoring of Coupled Tank Accompanied by PLC-OPC-MATLAB Architecture

Tushar Bhaskarwar*, Rajan Chile*, Sumit Aole*, Irraivan Elamvazuthi†
*Department of Instrumentation Engineering,
Shri Guru Gobind Singhji Institute of Engineering and Technology,
Vishnupuri, Nanded, Maharashtra,
Nanded, 431606, Maharashtra, India
†Department of Electrical and Electronics Engineering,
Universiti Teknologi PETRONAS (UTP),
Seri Iskandar, 31750, Perak, Malaysia

Email: bhaskar.tushar90@gmail.com (Tushar Bhaskarwar)
Email: rhchile@sggs.ac.in (Rajan Chile)
Email: sumit.aole@gmail.com (Sumit Aole)
Email: irraivan_elamvazuthi@utp.edu.my (Irraivan Elamvazuthi)
Corresponding author: Irraivan Elamvazuthi

***Abstract*—This paper focuses on one of the important aspects in the industry 4.0 revolution which is remote monitoring of industrial process through the Industrial Internet of Things (IIoT). In this paper, the cascade control application of coupled level process plant is considered for remote monitoring with IIoT concept. The level of the tank was regulated using PLC-OPC-MATLAB configuration. In order to control the level of a Coupled tank process, two controllers for primary and secondary loops are used to achieve the desired set point. Proportional Integral Derivative-Proportional (PID-P) control strategy is implemented in primary and secondary loop respectively for tracking the set point with minimum settling time. The ThingSpeak platform, a cloud-based server is used for remote monitoring purpose. Finally, the results of the level response and its online observation shown in the result section.**

***Index Terms*—MATLAB, Programmable Logic Controller (PLC), Remote monitoring, Internet of Things (IOT)**

I. Introduction

With more paradigms leading to advances in how products are designed, tailored and produced, the Internet is changing the manufacturing floor. Present technologies, such as cloud-based monitoring, the Industrial Internet-of-things (IoT), big data, Internet of services (IoS) and smart manufacturing are driving the emergence of the "Industry 4.0" i.e, “Fourth Industrial Revolution” as coined in German [1]. In home automation systems, the automotive sector, the agricultural sector and the health industry, cloud-based monitoring systems are important and efficient [2].

This cloud-based control and monitoring is applicable for industrial applications also. The industrial automation has been integrated using Industrial Internet of Things (IIoT) concept [3]. IIoT have been implemented using a well-known tool for researcher named as MATLAB [4]. In current industrial situation various complex control design need to be implemented which needs, MATLAB as software interface for various hardware. So, this software tool is used for implementation environment. MATLAB has inbuilt ThingSpeak server tool box in its Simulink library which is used for remote monitoring [5]. Some study was carried out for easy understanding of communication of MATLAB with industrial controller and IoT structure. Industrial devices can be connected to each other for IIoT purpose [6] [7] [8] [9] [10].

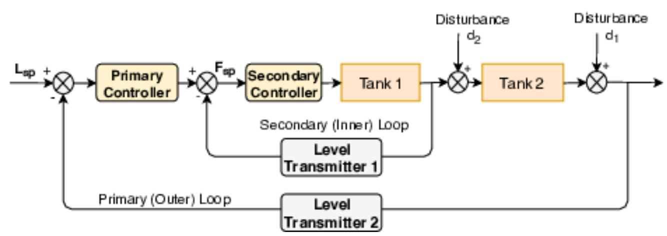

Fig. 1. Block diagram of cascade loop

The flexibility of control automation application can be make achievable according to user principles and this can be made possible using cloud based server which facilitate the development of remote monitoring automation applications in the form of physical mashups. Data logging function on cloud server can facilitate future data analysis, this all make implementable by using real time monitoring [11].

Monitoring systems let us aware the unexpected results occurred in system. Some hardware and software solutions are available for implementing monitoring systems which are very costly. The maintenance and installation of such systems are too expensive than the cost of the sensors themselves [12]. This paper deals with same concept of monitoring only the cost of all remote monitoring architecture is reduced by using free cloud database server.

The rest of this paper arranged as follows. Section 2 deals with methodology used in this paper for cascade level process controller with its mathematical modeling and its controller design. In section 3 steps regarding remote monitoring are discussed along with implementation of OPC server in PLC and MATLAB. Experimented results and discussion are explained in section 4. Finally, some conclusions are taken out from this paper.

II. Methodology

In many industries, for complex control operations practitioner prefer Cascade control (CC) structure instead of simple closed loop feedback structure. CC is used to improve the dynamic response of a feedback control loop and disturbances in the manipulated variable [13]. CC is often used in process industries and has applications in the field of automobile, robotics etc. The CC strategy is shown in Figure 1. This cascade control strategy is subdivided in two categories of control i.e., primary control (slow dynamics) and secondary control (fast dynamics). CC is also useful when actuator exhibits nonlinear behavior. In this paper, CC structure is validated for level-level control (LLC), and this is useful for the practitioners who aim to design and implement CC systems. This Study suggests that the primary loop should fed with PID, PI control action (to eliminate the offset generated by inner loop+outer loop) and secondary loop should fed with P controller.

In cascade control, the response of secondary loop is faster than response of primary loop so, it is necessary to control secondary loop (SL) faster than primary loop (PL) by making time constant of secondary loop greater than time constant of primary loop. In LLC process, offset in secondary loop is handled by outer loop PID controller this is the reason we feeding P controller in inner loop intentionally and making it faster than outer loop since here main aim is to control outer loop controlled variable.

A. *Cascade Process Description*

This is a two input and single output process. It is preferred to use CC strategy as the process involves interaction of level in tank 1 with level in tank 2 as shown in Figure 2. The cascade process has two cuboid tanks of equal size with pump as a final control element operated by variable frequency drive (VFD) as shown in Figure 2.

Tank 1 (Q1) is supplied with water from the reservoir and tank 1 is used as an inlet for tank 2 (Q2). Water fed back to the reservoir via the second tank outlet. Tank 1 level is measured by Level transmitter 1 (LT1) and Tank 2 level is measured by Level transmitter 2 (LT2) in 4 to 20 mA signal range. The height of level in tank 2, i.e. h2, is the required process variable (PV).

B. *Mathematical modeling*

The time rate of change of the liquid level in each tank is given by,

$$\dot{L}_i(t) = \frac{1}{A_i}(F_{in} - F_{out}) \tag{1}$$

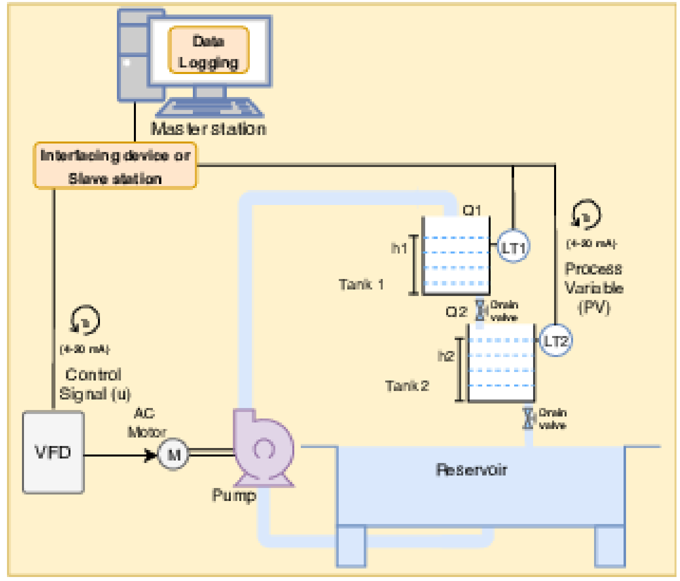

Fig. 2. Schematic of process setup

Where, $i = 1, 2$ and $L_i(t)$, A_i, F_{in}, and F_{out} are the liquid level (%), cross-sectional area, inflow rate and outflow rate respectively.

The volumetric flow rate input to tank 1 is directly proportional to the pump voltage which is handled by variable frequency drive, and it is estimated as,

$$F_{in}(t) = K_p u(t) \tag{2}$$

Where, K_p and $u(t)$ are the pump constant and pump frequency. The outflow velocity from the orifice at the bottom of each tank is computed using Bernoulli's law [14].

$$v_{out1}(t) = \sqrt{2gL_1(t)} \tag{3}$$

Where, g is the gravitational acceleration. Then, the outflow rate of tank is estimated as,

$$F_{out1}(t) = a_1\sqrt{2gL_1(t)} \tag{4}$$

where, a_1 is the cross-sectional area of the outflow orifice at the bottom of the first tank. The inflow of tank 2 is obtained as,

$$F_{in2}(t) = F_{out1}(t) \tag{5}$$

Using mass balanced principles and Equation (1) to Equation (5), dynamic equations of the liquid level in the two tanks can be written as follows:

$$\dot{L}_1(t) = \frac{K_p u(t) - a_1\sqrt{2gL_1(t)}}{A_1} \tag{6}$$

$$\dot{L}_2(t) = \frac{a_1}{A_2}\sqrt{2gL_1(t)} - \frac{a_2}{A_2}\sqrt{2gL_2(t)} \tag{7}$$

The plant model is also modeled by system identification toolbox in MATLAB into transfer function representation as given in Equation (8) (Secondary loop) and Equation (9) (Primary loop).

$$G(s) = \frac{0.88}{43.50s + 1} \tag{8}$$

$$G(s) = \frac{0.78}{89.66s + 1}e^{-8s} \tag{9}$$

C. Cascaded PID-P Controller

PID controllers are commonly used in industry and a large factory may have thousands of them, in instruments and laboratory equipment. It is error based control design and this is the main advantage of it, over other model based control design. PID control is by far the most common way of using feedback in natural and man-made systems. In engineering applications the controllers appear in many different forms: as a standalone controller, as part of hierarchical, distributed control systems, or built into embedded components. Most controllers do not use derivative action. The PID control and its implementation inside MATLAB is studied in this paper [15].

The generalized PID equation can be written by Equation (10),

$$u(t) = K_{pr}e(t) + K_i \int_0^t e(t)\,dt + K_d \frac{de(t)}{dt} \tag{10}$$

where, K_{pr}, K_i and K_d, all positive values, denotes the coefficient for the proportional, integral and derivative expression respectively. It can be also written as Equation (11)

$$u(t) = K_p(e(t) + \frac{1}{T_i}\int_0^t e(t)\,dt + T_d\frac{de(t)}{dt}) \tag{11}$$

where, $u(t)$ is the control signal and $e(t)$ is the control error (e = ref − y). The reference value, ref, is also called the setpoint. The control signal is sum of three terms: a proportional term that is proportional to the error, an integral term that is proportional to the integral of the error, and a derivative term that is proportional to the derivative of the error. The controller parameters are proportional gain K_{Pr}, integral gain K_i, derivative gain K_d, Integral time constant T_i, and derivative time T_d. The gains of PID-P loop are tuned on MATLAB Simulink environment and considered as final gain values. The proportional action acts on the present value of the error and works directly with respect to errors, the integral action represents an average of past errors, offset produced by proportional will get canceled by integral action and the derivative action can be interpreted as a prediction of future errors based on linear extrapolation.

Both PID-P controller are tuned in such a way that it should give as minimum as settling time for level in second tank and accordingly both controllers are tuned on Simulink.

III. Industrial Internet of Things (IIoT)

Industrial setup can be realized using MATLAB and PLC and its remote monitoring is experimented through various architectures. In current industrial reconstruction in technology, many challenges arises for developing remote monitoring of

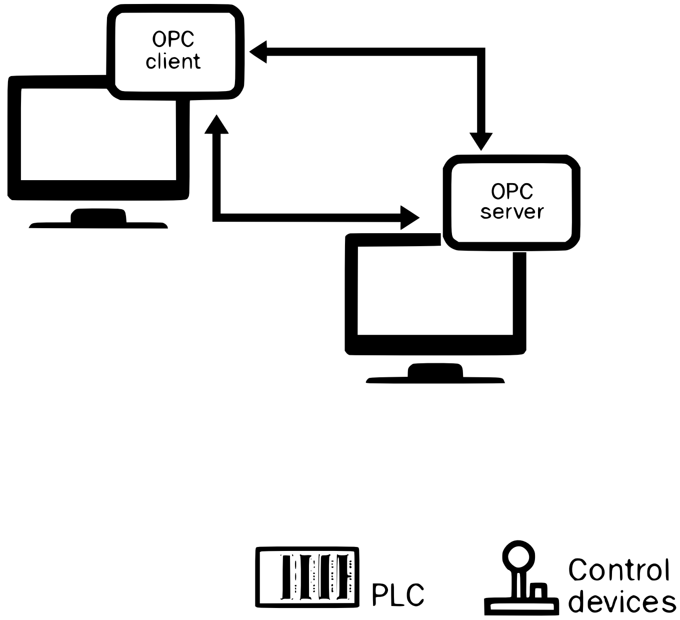

Fig. 3. OPC Architecture

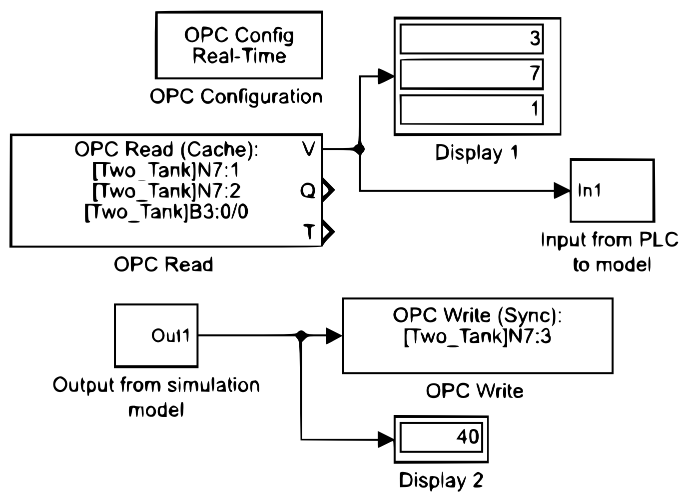

Fig. 4. OPC Configuration of MATLAB

processes. Cloud based control method is possible but certain amount of risk also came in various way. To minimize this risk we have only chosen to monitor the process by PLC-OPC-MATLAB configuration. This monitoring provides easy access of process parameters. Monitoring can also be done thorough smart mobile or handheld devices. Internet of things uses a cloud platform for logging its data on cloud server.

A. OPC Server

OPC (Object link enabled for Process Control) is named as the open process control protocol. It is an interoperability standard for the reliable and secure exchange of data in the industrial automation arena and in other industries. It is platform independent and ensures the seamless flow of information among devices from various vendors. The OPC Foundation is responsible for the maintenance and development of this standard [16]. Industrial controllers can be interchange there data through OPC server with other software or same other controllers. It is behaves as a language translator between two or more than two controllers. The reason of increase in adaptation of OPC is that, because of its less cost and its easy implementation. It provides good flexibility to users. Figure 3 represents the general architecture of OPC Protocol.

RSLinx software is used for configuring PLC as new DDE/OPC connection with MATLAB OPC server. For setup this arrangement, we have use topic configuration option in RSLinx software and select the PLC program which need to be control through OPC server of MATLAB. If perfect OPC communication established between PLC and MATLAB then all variables from selected PLC program will get appear for use in MATLAB environment. Some steps are need to be performed for perfect communication between PLC OPC server and MATLAB OPC server.

1) Select data source of PLC from RSLinx OPC server through topic configuration option.
2) Type command in MATLAB as "opcregister('~silent')" for Initializing OPC server of MATLAB.
3) Now, In Simulink take "OPC-Configuration" block from its library.
4) Open "OPC-configuration" and choose "RSLinx OPC" server for getting data from PLC to MATLAB and insert "OPC-Read" and "OPC-Write" block in it.
5) Use "OPC-read" block for feeding data to MATLAB from PLC and Use "OPC-write" block for sending to PLC from MATLAB.

Follow above procedure to get all variables from expected PLC program to MATLAB OPC client. Now inside "OPC Read" one should write [OPCfoldername]F8:1, for getting data of #F8:1 variable of PLC. Similarly, it can be written for "OPC Write" block. The typical example of OPC configuration is shown in Figure 4.

B. Remote Monitoring

In this paper PID-P controller is designed on MATLAB Simulink environment and the process variable coming from PLC are monitored on "ThingSpeak" cloud server provided by MathWorks Inc. For monitoring on "ThingSpeak" cloud server first we need to open account on ThingSpeak platform. Steps for communication between ThingSpeak server and MATLAB.

1) MATLAB version 2017b is used in this experiment validation because this version gives inbuilt "ThingSpeak Output", "ThingSpeak Input" channels block in library of SIMULINK.
2) Only "ThingSpeak Output" block used in this study for monitoring purpose. This block need to be configured by Write API Key, Channel ID and Update period time.
3) For getting continuous data on "ThingSpeak" platform we need to Run the SIMULINK program for "infinite" period of time.

Typical Configuration of all "ThingSpeak server" connection setting is shown in Figure 5.

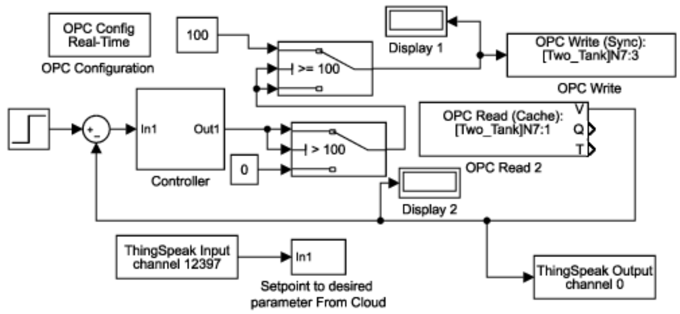

Fig. 5. ThingSpeak Connection

Configure channel parameters icon in ThingSpeak output block gives access to enter channel ID, write API key and update period. The main advantage of this server is that, it is totally free and accessible to all authentic viewers.

Two keys are available on ThingSpeak server as shown in Figure 6. Write API key and Read API key, used to write a data on ThingSpeak server and read the data from ThingSpeak server respectively. Real time data monitoring is shown in Figure 9.

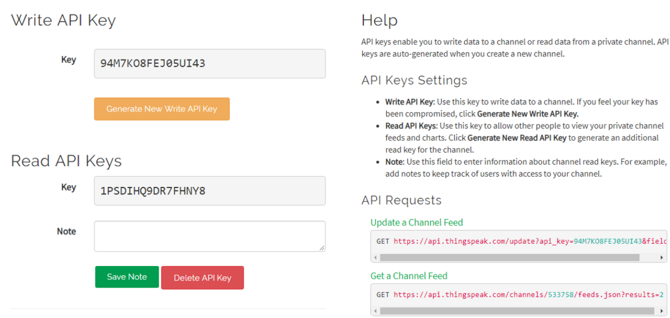

Fig. 6. ThingSpeak API Key

IV. Result and Discussion

The proposed remote monitoring structure is validated on real plant with PID-P controller on MATLAB environment. Setpoint of 40% was given and its level response for both tank 1 and 2 are monitored on MATLAB as well as on ThingSpeak cloud server as shown in Figure 7 and Figure 9 respectively. The control signal required for maintaining 40% level was acquired inFigure 8. It was observed that, the communication established between PLC and MATLAB gives very smooth control and fewer sampling time. It means the OPC connection between PLC and MATLAB works efficiently with sampling time of 3 m se

It was observed that the data on cloud server was updated at every 14 seconds and it is represented by one dot as shown in Figure 9. Just for sake of simplicity only one parameter was monitored on cloud platform. As per user application, one can monitor number of parameter at a time.

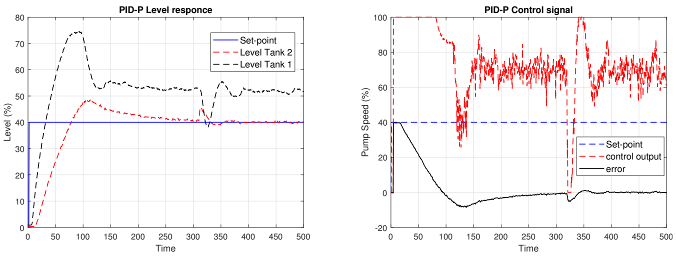

Fig. 7. Level response

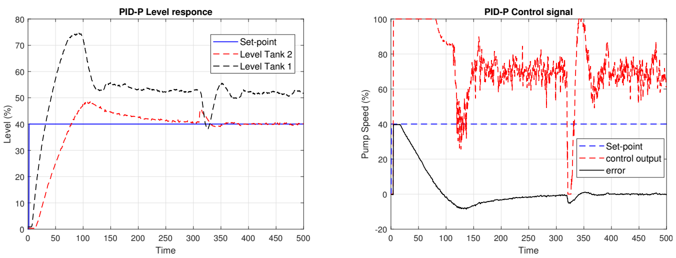

Fig. 8. Control signal

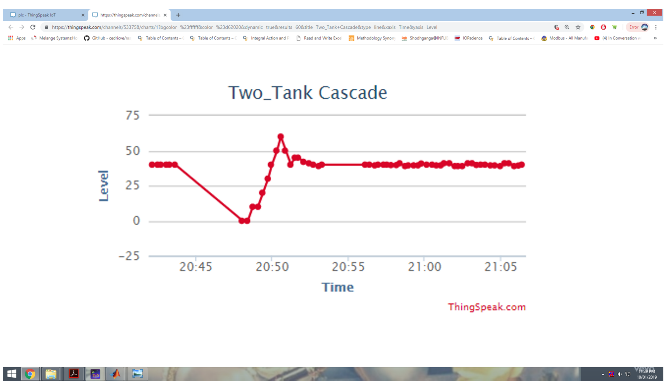

Fig. 9. Online remote monitoring

V. Conclusion

Thus, the Internet of things can be implemented through PLC-OPC-MATLAB configuration and it is validated on coupled tank. PLC and MATLAB connection was seen to be very fast and sampling time was near about 1 msec. This study provides very simpler procedure to implement remote monitoring using MATLAB and ThingSpeak server. PID-P Controller gives good result for coupled tank system.

Acknowledgment

The authors would like to thank SGGSIE & T, Nanded, India and Universiti Teknologi PETRONAS(UTP), Malaysia for their support.

References

[1] H. Kagermann, W. Wahlster, and J. Helbig, "Recommendations for implementing the strategic initiative INDUSTRIE 4.0," *ACATECH NATIONAL ACADEMY OF SCIENCE AND ENGINEERING*, 2013.

[2] K. E. Adetunji and M. K. Joseph, "Development of a Cloud-Based Monitoring System Using 4Duino: Applications in Agriculture," in *2018 International Conference on Advances in Big Data, Computing and Data Communication Systems (icABCD)*, Aug 2018, pp. 4849–4854.

[3] D. Miller, "Blockchain and the Internet of Things in the Industrial Sector," *IT Professional*, vol. 20, no. 3, pp. 15–18, May 2018.

[4] "MATLAB." [Online]. Available: https://in.mathworks.com/

[5] "ThingSpeak Cloud Server." [Online]. Available: https://thingspeak.com/

[6] O. Givehchi, J. Imtiaz, H. Trsek, and J. Jasperneite, "Control-as-a-service from the cloud: A case study for using virtualized PLCs," in *2014 10th IEEE Workshop on Factory Communication Systems (WFCS 2014)*, May 2014, pp. 1–4.

[7] "Internet of Things." [Online]. Available: https://en.wikipedia.org/wiki/Internet_of_things

[8] E. Sisinni, A. Saifullah, S. Han, U. Jennehag, and M. Gidlund, "Industrial Internet of Things: Challenges, Opportunities, and Directions," *IEEE Transactions on Industrial Informatics*, vol. 14, no. 11, pp. 4724–4734, Nov 2018.

[9] L. D. Xu, W. He, and S. Li, "Internet of Things in Industries: A Survey," *IEEE Transactions on Industrial Informatics*, vol. 10, no. 4, pp. 2233–2243, Nov 2014.

[10] H. S. Raju and S. Shenoy, "Real-time remote monitoring and operation of industrial devices using IoT and cloud," in *2016 2nd International Conference on Contemporary Computing and Informatics (IC3I)*, Dec 2016, pp. 324–329.

[11] B. Cheng, S. Zhao, S. Wang, and J. Chen, "Lightweight Mashup Middleware for Coal Mine Safety Monitoring and Control Automation," *IEEE Transactions on Automation Science and Engineering*, vol. 14, no. 2, pp. 1245–1255, April 2017.

[12] F. Salvadori, M. de Campos, P. S. Sausen, R. F. de Camargo, C. Gehrke, C. Rech, M. A. Spohn, and A. C. Oliveira, "Monitoring in Industrial Systems Using Wireless Sensor Network With Dynamic Power Management," *IEEE Transactions on Instrumentation and Measurement*, vol. 58, no. 9, pp. 3104–3111, Sep. 2009.

[13] R. P. Krishnaswamy, P. G. Rangaiah, R. K. Jha, and P. B. Deshpande, "When to use cascade control," *Industrial & Engineering Chemistry Research*, vol. 29, no. 10, pp. 2163–2166, 1990.

[14] O. K, "System Dynamics." Englewood Cliffs, NJ: Prentice-Hall., 1998.

[15] K. J. Aström and T. Hägglund, "PID Controllers: Theory, Design, and Tuning, Second Edition." America: Research Triangle Park, NC: Instrument Soc., 1995.

[16] M. Schleipen, "OPC UA supporting the automated engineering of production monitoring and control systems," in *2008 IEEE International Conference on Emerging Technologies and Factory Automation*, 2008.