2018 International Conference on Computer and Applications (ICCA)

A High Availability Networked Control System Architecture for Precision Agriculture

Hassan Ibrahim, Norhan Mostafa,
Hassan Halawa, Malak Elsalamouny,
Ramez Daoud, Hassanein Amer
Elect. & Comm. Eng. Dept.
The American University in Cairo
Cairo, Egypt
hassanibrahim@aucegypt.edu
hhhalawa@ieee.org, rdaoud@ieee.org

Amr Shaarawi
Physics Department
The American University in Cairo
Cairo, Egypt
shaarawi@aucegypt.edu

Ahmed Khattab, Hany ElSayed
Electronics and Electrical
Communications Engineering
Department
Cairo University
Cairo, Egypt
akhattab@ieee.org, helsayed@ieee.org

Abstract— **Precision agriculture systems are usually studied in the context of Wireless Sensor Networks. In this paper, a hierarchical Networked Control System (NCS) is proposed for two Greenhouses. This system is built on top of switched Ethernet and Wi-Fi, and is IoT-based. Fire and CO2 sensors imposed a 1s real-time deadline. It is shown how to allocate Wi-Fi channels to prevent interference in this relatively large NCS. Riverbed simulations proved that the two-Greenhouse NCS did not suffer any packet loss and was able to meet real-time constraints. Fault Tolerance at the controller level is then added to the system. Riverbed simulations again proved that the system can operate as a 1-out-of-2 system; if one controller fails, the other controller can operate both greenhouses. A case study is then presented to show that fault tolerance can decrease downtime which is a very important advantage especially in developing countries.**

Keywords—NCS, precision agriculture, greenhouse, Ethernet, Wi-Fi, fault tolerance, reliability, IoT

I. INTRODUCTION

Greenhouses have a vital role in modern precision agricultural systems. They provide the ability to control agricultural parameters and conditions to create the adequate atmosphere for different crops. Accordingly, this provides the capability to overcome the environmental and seasonal restrictions in plantation, and produces crops with finer quality.

The agricultural practices inside the greenhouse are in continuous evolvement, starting by deploying different sensor nodes inside the greenhouse. These sensor nodes replace the human observations in measuring the environmental conditions and parameters inside the greenhouse in order to get more accurate measurements with the desired sampling frequency [1, 2]. Furthermore, IoT contributes to the evolvement of the greenhouse agriculture [3]. The information about monitored parameters using the sensor nodes inside the greenhouse can be analyzed, stored over the cloud, and presented over any Internet enabled device [4]. Also, the greenhouse enhancement goes beyond monitoring and reporting the environmental parameters; it extends to reach the remote control of different greenhouse conditions [5]. Based on the information about the greenhouse that is collected using the monitoring system, decisions can be taken about what should be done inside the greenhouse [6].

However, in previous works, small to medium systems based on Wireless Sensor Networks (WSNs) were studied for proof of concept. There is no practical definition for a system architecture that is ready for implementation on a relatively large scale with a guaranteed efficiency and reliability. In this paper, the proposed greenhouse architecture is studied as a hierarchical distributed Networked Control System (NCS) [7, 8]. This is appropriate since the system is composed of a large number of sensors, controllers and actuators interconnected by a shared network that is designed to carry small packets and meet real-time control constraints with minimal packet loss and high reliability. This NCS is built on top of both Ethernet and Wi-Fi. Furthermore, the proposed system is relatively large since it consists of two Greenhouses. It will be shown how to find a channel allocation that enables this relatively large system to meet all NCS real-time constraints such as packet loss and delay [9-12]. Next, fault tolerance is investigated. Both identical greenhouses are connected over the same network and it is proven that they can still meet system requirements even if one controller fails. It will be shown that this fault-tolerant architecture increases system availability, which is very important especially in developing countries.

The rest of this paper is organized as follows. In Section II, the proposed system architecture is explained. Simulation results are discussed in Section III and Fault Tolerance is investigated in Section IV. Finally, the paper is concluded in Section V.

II. PROPOSED SYSTEM ARCHITECTURE

In this paper, two Greenhouses are studied in the context of fault-tolerant NCSs, with dimensions 200m x 40m each, and divided into 5 square cells (40m x 40m). Each Greenhouse can be viewed as an NCS with a controller, smart sensors and the actuators. The two NCSs in the two greenhouses are connected to the same network in order to add fault tolerance to the system. This is one of the major contributions of this work. This architecture is also IoT-based for monitoring and remote control. It is composed of three hierarchical layers: the Sensor/Actuator frontend Layer that has the smart sensors and smart actuators, the Data Management Layer that consists of the Greenhouse controller and finally, the Cloud-based Backend Layer.

The Sensor/Actuator frontend layer is composed of a group of sensor nodes and actuators distributed in each greenhouse cell. A sensor node is a simple microcontroller that hosts a collection of 9 sensors with different data rates according to the

978-1-5386-4371-6/18/$31.00 ©2018 IEEE

criticality of the measured parameter. Those sensors measure 9 different environmental parameters: Temperature, humidity, light, soil moisture, salinity, and dew; they are sampled at a data rate of 1 byte every 30 seconds. Pesticide sensors are sampled for 1 byte every 5 seconds, and the most critical sensors: fire and CO2 are sampled with 1 byte every second. The microcontroller for these sensor nodes is equipped with a Wi-Fi interface in order to transmit the collected data wirelessly to a local access point for the greenhouse cell which relays the collected data to the Greenhouse controller. The wireless protocol is IEEE-802.11n with a transmit power of 5mWatts and frequency band of 5GHz since it supports a higher number of channels than those from 2.4GHz. Due to the abundance of channels, each cell within the same greenhouse uses a different frequency channel to eliminate interference. Table I shows the channel assignment for both Greenhouses. Also, four cameras are placed in the corners of each cell to capture live video with a resolution of 5MP and a transmission rate of 12 FPS. Each camera is connected to the controller of the cell via an Ethernet cable in order to reduce the interference between its high rate traffic and the traffic of the sensor nodes.

TABLE I. WI-FI CHANNEL ASSIGNMENT ID

Channel Assignment	Cell 1	Cell 2	Cell 3	Cell 4	Cell 5
Greenhouse 1	56	52	48	44	40
Greenhouse 2	36	60	64	149	153

Four actuators of the Sensor/Actuator frontend layer are located in each greenhouse cell in order to control the lighting, irrigation, fans, and curtains inside the cell; in addition, there is a fire extinguisher actuator for the entire Greenhouse. Similar to the sensor nodes, the actuators execute actions at different rates based on the criticality of the action. All actuators update actions every 30 seconds except for the fire actuator which takes action every 1 second. The actuator nodes are connected to the controller of the respective cell using Ethernet cables in order to receive the control action commands from the Greenhouse controller.

The data management layer is the greenhouse controller that collects the data from the sensor nodes of the frontend layer, processes such data locally and accordingly takes appropriate control actions inside the premises of the greenhouse. The control action is passed to the actuators of the frontend where it is implemented.

The two Greenhouse controllers share periodical watchdog signals to ensure each one’s functionality. If a failure occurs in one of them, the data acquired from the greenhouse of the failing controller is forwarded to the other operational controller. Thus, this controller takes the required actions for both greenhouses simultaneously. This increases the reliability of the greenhouse system as will be shown later.

A cloud-based backend layer allows the data of the monitored greenhouses to be accessible through the Internet. The interface to this backend is implemented through an Internet gateway node. Each microcontroller node relays collected sensor data to this gateway. The gateway then relays the information (after analyzing it and creating aggregate reports) to a cloud server for storage and extensive data analysis. The Internet gateway also forwards actuation requests from the cloud server to the microcontrollers. Thus, through Internet access, it is possible to receive a complete picture of all the information and actions taken inside the greenhouses and possibly override the controller actions remotely. The backend cloud server facilitates the end-users' ability to access the sensed data and control the actions if needed. These objectives are achieved by implementing a set of services such as data storage, data analytics, data security, and data visualization.

Using Riverbed Modeler [13], the proposed system is simulated where the two greenhouses are placed horizontally beside each other. The greenhouse's main controller is connected over Ethernet to 1) the access points of the cells 2) the Internet Gateway 3) the controller of the neighboring greenhouse as shown in Fig. 1. Fig. 2 illustrates the greenhouse cell.

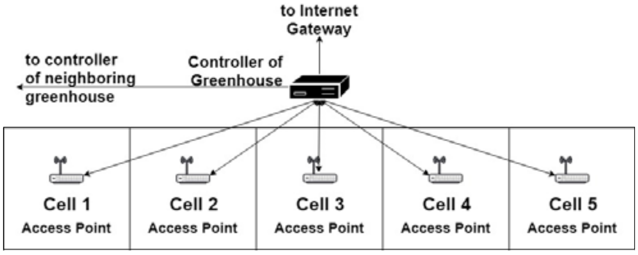

Fig. 1. Greenhouse schematic

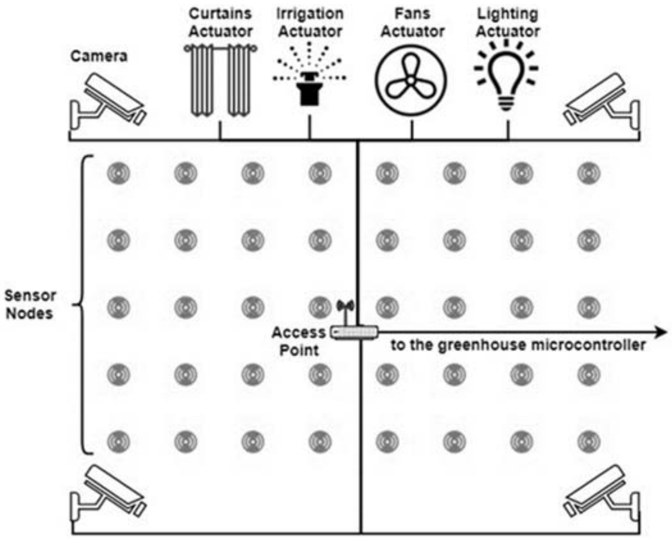

Fig. 2. Greenhouse cell

A. Simulation Scenarios

In order to validate the functionality and reliability of the proposed architecture, three different simulation scenarios are performed to cover all the expected situations during the operation of the greenhouse. The first scenario is the Fault-Free scenario where the main controllers of the two greenhouses are functioning properly. The other two scenarios are the Controller Failure scenarios, where the controller of one of the two greenhouses goes out of service in each scenario.

Fault-Free Scenario: In this scenario, the sensor nodes in each of the greenhouses are sending to their respective controller only. This setup represents the daily operation of the greenhouses without any failures. In such a setup, each main controller is operational and is receiving data from all sensors and cameras in addition to distributing commands to the actuators in the greenhouse. Additionally, a watchdog signal of 1B is propagated between the two main controllers at 1 second intervals (to acknowledge each other's ability to operate properly).

Controller Failure Scenarios: In these scenarios, the system is simulated when one of the two controllers fails. The other functioning greenhouse controller takes over the operations of both greenhouses. In this setup, sensors and cameras of both greenhouses send their data to the functioning controller. Similarly, the control action to the actuators of the two greenhouses is sent from the functioning greenhouse controller.

B. Performance Evaluation Metrics

The complete information cycle inside the greenhouse starts from the sensor nodes when it sends the collected information to the main controller of the greenhouse, and ends at the actuators that receive the required action commands from the controller. Accordingly, in order to be able to evaluate the performance of the proposed system architecture, the metrics that indicate whether the different simulation scenarios are performing as desired or not, are the Packet Loss and the Delay. Overdelayed packets in the information cycle are undesirable, because it will cause taking actions at incorrect timing; therefore the total delay for the information cycle has to be below the sampling period of the fastest part of the system, which is the fire and CO2 systems (the fire and CO2 sensors send information every 1 second and the fire extinguisher actuator takes action every 1 second). Hence, all delays must be below 1 second.

III. SIMULATION RESULTS

For each scenario, simulations were performed with 33 seeds for 1800 seconds, and the maximum delays and packet losses for each seed were considered and analyzed with 95% confidence. The simulation showed that there is no packet loss at any point of time during the simulation of the three scenarios over their 33 seeds. Also, the measured end-to-end delay inside the greenhouse includes packet transmission, propagation, processing, and queuing delay. Table II has the results for the fault-free scenario while Table III has those for the scenario where the controller of the second Greenhouse has failed. If the other controller fails, simulations results are very similar to those in Table III. The values in both tables are the minimum and the maximum of the 95% confidence analysis for the amounts of delay, performed for the maximum delay incident per seed for the 33 seeds of each scenario.

For the Fault-Free scenario, symmetrical amounts of delay are observed for the two greenhouses because the information takes similar paths in both greenhouses with no dependency for any of the greenhouses on the other (see Fig. 3). For the controller failure scenarios, higher delays (specially between sensors and the operational controller) are observed in many runs in the greenhouse with the failing controller due to the additional delay of forwarding data to the controller of the neighboring greenhouse, and retrieving the commands to be executed by the actuators.

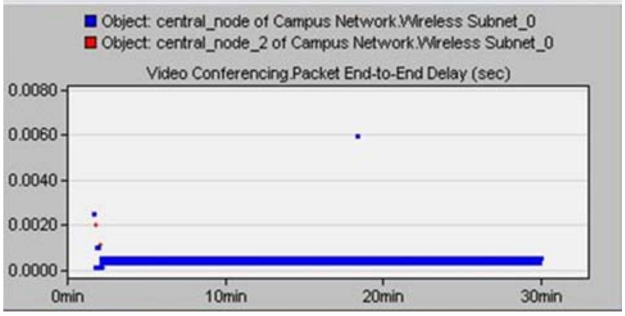

Fig. 3. Delay from sensor to controller in both greenhouses for one seed in error-free scenario

It is clear that all delays are below the 1 second constraint as explained above. Along with the fact that there is zero packet loss, this proves that this fault-tolerant NCS system succeeds in meeting system requirements.

TABLE II. FAULT-FREE SCENARIO DELAY

Fault-Free Scenario Total Delay	**msec**
Greenhouse 1	[5.6; 6.0]
Greenhouse 2	[5.6; 5.9]

TABLE III. CONTROLLER FAILURE - SCENARIO 1 DELAY

Controller Failure Scenario Total Delay	**msec**
Greenhouse 1	[5.5; 5.8]
Greenhouse 2 (Failing Controller)	[5.5; 5.9]

IV. RELIABILITY ANALYSIS

It is clear from the previous section that the control function of the two-greenhouse system is fault-tolerant. In other words, the control function is implemented by a 1-out-of-2 fault-tolerant architecture. Fault tolerance is expected to reduce system downtime.

In general, a failed controller may cause damage to plants cultivated in greenhouses. The extent of such damage depends on the type of grown crops. For example, ornamental plants and flowers can be very sensitive to variations in humidity and temperature, and delayed or no actions can cause significant financial losses. The same applies to certain types of vegetables and seedlings grown in greenhouse nurseries. For these reasons, downtime is very costly. Especially in developing countries, repair times can be relatively high because spare parts are not usually stored on site or even at the dealer's premises; they have to be imported and customs are rarely predictable.

Next is a quantitative study to illustrate the effect of fault tolerance on system reliability. As mentioned above, the system under study is a 1-out-of-2 system. Let the time to failure be exponentially distributed. Hence, the failure rate is constant [14]. Let $\lambda 1$ be the failure rate of the controller in the first greenhouse (K1) and $\lambda 2$ be the failure rate of the controller in the second greenhouse (K2). Fig. 4 has the Markov model describing the behavior of this 1-out-of-2 system. In state G1G2, both greenhouse controllers are operational. If K1 fails, the system moves to state G2 and both greenhouses are operated by K2. This transition has a rate λ1. While in state G2, K1 is being repaired at a rate μ. The same reasoning is true in case K2 fails; the system moves to state G1 at a rate λ2. While in state G2, if K2 fails before K1 is repaired, the two-greenhouse system fails completely (state SF). Note that, in Fig. 4, the system moves back from state SF to state G1 at a rate μ (assuming a single repair person [14]). The same is true for the transition from SF to state G2. This Markov model can be solved using the techniques in [14].

$$\frac{dP}{dt} = P \times T$$

where P is the vector of state probability functions and T is the differential state-transition rate matrix.

$$T = \begin{bmatrix} -\lambda 2 - \lambda 1 & \lambda 2 & \lambda 1 & 0 \\ \mu & -\lambda 1 - \mu & 0 & \lambda 1 \\ \mu & 0 & -\lambda 2 - \mu & \lambda 2 \\ 0 & \mu & \mu & -2\mu \end{bmatrix}$$

The Steady State Availability (AVss) is the steady state probability of **not** being in the failed state SF.

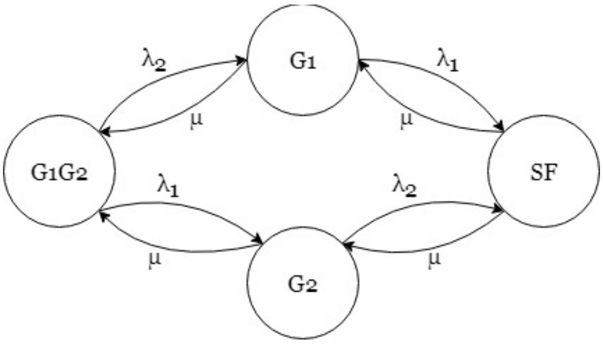

Fig. 4. Markov model

Next is a simple case study to illustrate the advantage of using this fault-tolerant scheme. Let λ1 = (1/6) month^{-1} and λ2 = (1/8) month^{-1}. Furthermore, let μ = 2 month^{-1}. Using the software tool SHARPE [15], the Steady State Availability (AVss) can be obtained. It is equal to 99.547%. If the spare parts are stored on site, μ will increase. Let μ=30 month^{-1}. AVss will increase to 99.9977%. If fault tolerance is not applied to the two greenhouses, AVss for the first Greenhouse and the second Greenhouse would be 92.3% and 94.1% respectively if μ=2 month^{-1}. If μ=30 month^{-1}, AVss will increase to 99.45% and 99.59%. Hence, when spare parts are stored on site, fault-tolerance will slightly increase AVss; however, if spare parts have to be imported after the occurrence of a failure, fault tolerance has a significant effect on AVss. This case study indicates that if it is difficult to store spare parts, the proposed fault-tolerant architecture will guarantee a very high AVss.

V. CONCLUSION

The development of NCS for greenhouses is an interesting topic for research and development. Furthermore, the integration of the IoT solutions empowers the greenhouse NCS.

In this paper, a system architecture for an NCS inside the greenhouse connected to a cloud was proposed, and fault-tolerance was incorporated in the design. In the system simulation, two greenhouses were placed and the different scenarios were simulated to show that even if one of the controllers fails, the controller of the neighboring greenhouse operates for both greenhouses successfully. A confidence analysis was performed on the simulation results of the system simulation, and all the results showed that the system meets all the real-time constraints.

Finally, a case study is presented to show that this fault-tolerant greenhouse system would still be highly available even if spare parts are not stored on site and have to be imported.

REFERENCES

[1] M. Erazo, D. Rivas, M. Perez, O. Galarza, V. Bautista, M. Huerta and J.L. Rojo, “Design and implementation of a wireless sensor network for rose greenhouses monitoring”, Proceedings of the International Conference on Automation, Robotics and Applications, Queenstown, New Zealand, February 2015.

[2] M. Srbinovska, C. Gavrovski, V. Dimcev and A. Krkoleva, “Environmental parameters monitoring in precision agriculture using Wireless Sensor Networks”, Journal of Cleaner Production, May 2014.

[3] T. Guo and W. Zhong, “Design and implementation of the span greenhouse agriculture Internet of Things system”, Proceedings of the International Conference on Fluid Power and Mechatronics, Harbin, China, August 2015.

[4] T. Gomes, J. Brito, H. Abreu, H. Gomes and J. Cabral, “GreenMon: An efficient wireless sensor network monitoring solution for greenhouses”, Proceedings of the IEEE International Conference on Industrial Technology (ICIT), Seville, Spain, March 2015.

[5] H. Sampaio and S. Motoyama, “Implementation of a greenhouse monitoring system using hierarchical Wireless Sensor Network”, Proceedimgs of the IEEE 9th Latin-American Conference on Communications (LATINCOM), Guatemala City, Guatemala, November 2017.

[6] Q. Bai and C. Jin, “The remote monitoring system of vegetable greenhouse”, Proceedings of the International Symposium on Computational Intelligence and Design, Hangzhou, China, December 2017.

[7] X. Ge, Y. Fuwen and Q. Han, "Distributed networked control systems: A brief overview." Information Sciences, vol. 380, 2017, pp. 117-131.

[8] R.A. Gupta and M.Y. Chow, "Networked control system: Overview and research trends", IEEE Transactions on Industrial Electronics, Vol. 57, No. 7, July 2010, pp. 2527-2535.

[9] T. Skeie, S. Johannessen and C. Brunner, “Ethernet in substation automation,” IEEE Control Syst., Vol. 22, No. 3, 2002, pp.43-51.

[10] J.D. Decotignie, “Ethernet-based real-time and industrial communications , ” Proceedings of the IEEE, Vol. 93, No. 6, 2005.

[11] M. Felser, “Real-time Ethernet – Industry prospective,” Proceedings of the IEEE, vol. 93, no. 6, June 2005.

[12] R. Steigmann and J. Endresen, “Introduction to WISA: WISA – Wireless Interface for Sensors and Actuators,” ABB, July 2006.

[13] Official Website for Riverbed Modeler," [Online]. Available: https://www.riverbed.com/ Riverbed

[14] D.P. Siewiorek and R.S. Swarz, “Reliable computer systems design and evaluation,” A K Peters, Natick, Massachusetts, 1998.

[15] Official site for SHARPE: http://sharpe.pratt.duke.edu.