XVII th World Congress of the International Commission of Agricultural and Biosystems Engineering (CIGR)

Hosted by the Canadian Society for Bioengineering (CSBE/SCGAB)
Québec City, Canada June 13-17, 2010

DEVELOPMENT OF A SCADA SYSTEM FOR ACCESS, PROCESSING AND SUPERVISION OF DATA COMING FROM A WIRELESS SENSORS NETWORK IN AGRO-ENVIRONMENTAL APPLICATIONS

LUIS MANUEL NAVAS GRACIA[1], EDUARDO MARTÍN BRAVO[1], GONZALO RUIZ RUIZ[1], ADRIANA CORREA GUIMARAES[1], JAIME GÓMEZ GIL[2], FERNANDO GONZÁLEZ HERRERO[1], FÁTIMA BAPTISTA[3], VASCO FITAS DA CRUZ[2], MIGUEL DE CASTRO NETO[4]

[1] Department of Agricultural and Forestry Engineering, University of Valladolid. Avda. Madrid, 44, 34004 Palencia - SPAIN <lmnavas@iaf.uva.es>
[2] Department of Communications and Signal Theory and Telematics Engineering, University of Valladolid, Camino del Cementerio, s/n, 47011 Valladolid - SPAIN <jgomez@tel.uva.es>
[3] Department of Agricultural Engineering, University of Évora/ICAAM, ZIP 94, 7002-554 Évora – PORTUGAL <fb@uevora.pt>
[4] Instituto Superior de Estatística e Gestão de Informação, New University of Lisboa. Campus de Campolide, 1070-312 Lisbon – PORTUGAL <mneto@isegi.unl.pt>

CSBE101222 – 8th World Congress on Computers in Agriculture (WCCA) symposium

ABSTRACT. Wireless sensors networks appeared in the 1970's for military and industrial use. They have since undergone a major evolution, particularly since the 90's, thanks to the improvements in wireless communications. These changes have allowed them to participate in a wide variety of applications in different sectors such as agriculture and environment. This paper shows the development of a SCADA application programmed with LabVIEW® 8.6 (National Instruments), which allows management of data received by wireless sensors networks through a friendly interface for users. For the application shown in this paper we have worked with a MEP 510 sensors network (Crossbow). The functionalities implemented are the following: Network configuration; Data storage into database; Statistical processing of historical data with polynomial adjustment and spline interpolation; Visualization by data graphics in real time and historical data; Visualization of 2D intensity diagrams from the spatial distribution of sensors; and Creation of a users registry system that allows, depending on the category assigned, receiving or not access privileges in the application. As a complement we have developed the possibility of remote access. Sensors network implemented and the applications developed have been checked by operational tests for each functionality, as well as sensors joining and leaving the network situations, range of variables and working modes. The results obtained show the robustness of the SCADA application and the limitations of wireless sensors networks operating on field conditions.

Keywords: Wireless sensors network, database, LabVIEW, agro-environmental

INTRODUCTION. Wireless sensors networks have become a point of interest to all productive sectors due to its wide variety of application, having a widespread use and providing an almost unlimited number of sensors. The origin of distributed sensor networks is the late 70's, in military and industrial applications with low scaled wired sensors. However from the 90's there was an evolution of wireless technologies and the number of integrated transistors on a chip (VLSI technology, Very Large Scale Integration) which allowed the beginning of research in wireless sensors networks to large scale. Researchers at the University of Berkeley developed and embedded devices for a wireless network of sensors called motes (from remote). These devices due to its nature of being easily programmable, fully functional can provide a platform for research with a price not too high, contributing significantly to the revolution of wireless sensor networks (Krishnamachari, 2005).

The most common topologies for wireless sensors networks are the following:

1. **One hop star topology**: It is the easiest topology, each node connects directly to the gateway. It has two disadvantages: bad scalability and robustness, especially for huge extensions.
2. **Mesh or multihop grid topology**: Especially fit for huge extensions solving the problems of one hop star topologies. There are two different variations:
    a. Sensors ordered into rows and columns forming a 2D grid.
    b. Sensors placed randomly.
3. **Hierarchical topologies**: Based on decomposing the network into clusters managed independently. Each cluster can have one of the previous topologies or even be wired in some cases.

The main advantages for wireless sensors networks are: low cost, easy use for temporary locations (Aakvaag *et al*., 2006), good signal to noise relationship (Pottie e*t al.*, 2000) and energy saving in multihop networks (Intanagonwiwat *et al.*, 2000; Zhao *et al.*, 2004). But they also have some disadvantages: low life-cycle batteries, components with limitations (low speed, memory…) (Hill *et al.*, 2000) and networks with a huge number of sensors.

The applications for wireless sensors networks can be divided into two different groups:

1. **Classification by activity** (Yingshu *et al.*, 2008):
    a. **Detection of events and alerts**: Finding abnormal data or failures into processes. Those cases are not very common so sensors can be inactive most of the time, moving to an active situation when the alert occurs. To prevent false alarms more than one sensor can be used.
    b. **Data collection and periodic reminders**: Each sensor sends data periodically to be processed or stored on destination.
    c. **Search by destination**: The destination asks the sensors for the information they have instead of receiving it periodically. This way the destination has only to manage the information it considers necessary.
    d. **Tracking activities**: Registering movements of people or objects, either for security reasons or to create statistics to improve infrastructures.
2. **Classification by field** (National Instruments Developer Zone, 2009):
    a. **Medical**: Improving less invasive monitoring of patients, improving health care.
    b. **Basic services**: Such as electricity, public lighting, municipal waters etc. allowing data collection without a high cost, with low power consumption and resource optimization.
    c. **Remote monitoring**: It can be a complement of wired environments, avoiding the use of wire, reducing the budget and allowing new applications, including:

i. Environmental monitoring of air, water and soil.
ii. Structural monitoring for buildings and bridges.
iii. Industrial machinery monitoring.
iv. Processes monitoring.
v. Asset tracking.

Each sensor of the network can be defined as a device with a battery capable of detecting physical quantities (Yingshu *et al.*, 2008), having the following components (Aakvaag *et al.*, 2006; Krishnamachari, 2005):

1. **Low power embedded processor**: It has to process the information from that sensor and from the others.
2. **Memory/storage**: It can be divided into:
   a. Program memory: Having the instructions to be executed into the processor.
   b. Data memory: Having the information obtained by the sensor and local information.
3. **Radio transmitter/receiver**: Usually with a low data rate (10-100 kbps) and low distance range (<100 m.) but normally the component with the higher consumption of energy.
4. **Sensor**: There can be one or more. It works with low data rate and it can be of a wide variety: temperature, humidity, pressure…
5. **GPS**: It only appears in sensors where you need to determine its position because of its cost.
6. **Power supply**: Usually a battery.

OBJECTIVES. The first objective of this project is to find and study information about wireless sensors networks to achieve a global perspective on its features and general functionalities. The second and main objective of this project is the configuration and launching, interpretation and management of the information provided by a wireless sensors network of Crossbow for agro-environmental applications. For all of this, we use the manuals of the sensors (Crossbow MEP-SYS sensors users manual, 2007) which indicates fields that form the packets transmitted and may identify the different information (temperature, humidity ...) offered by the sensors. This will enable to perform appropriate conversions to make available information in the desired units. After obtaining useful information from the sensors, we must work with it to offer it to a user as simple as possible. The aim is that the user can interact with data on two different scenarios: one in real time and the other with historical data. The system also has to alert the user when a value not permitted is received, allowing to correct any problem in a few time. Finally the information received has to be stored to be checked in future times for reviews or analysis of working.

DEVELOPMENT.

Application structure. Motes are placed where we want to get the humidity and temperature parameters. Its information is sent by radio to the base station, made of a radio receiver and a MIB520 card with a USB connection for a computer. The application we have developed using the LabVIEW 8.6 programming environment is stored in the computer and receives data by the USB. It can also communicate by two .udl extension archives with two databases to store the data. The paths for these two archives are read from a file created automatically. All this structure can be seen schematically in Figure 1.

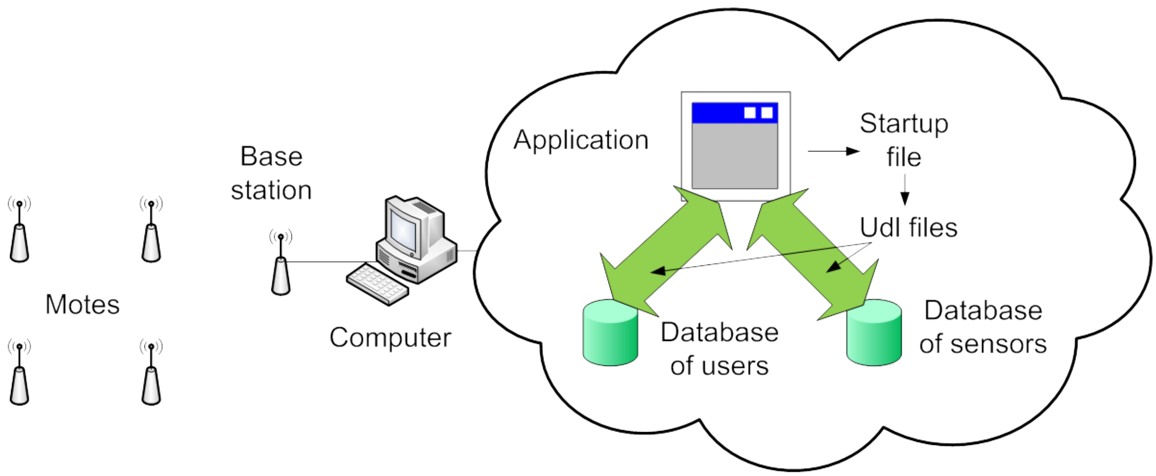

Figure 1. Scheme of the components of the system.

Databases. In the application we handle two databases: one containing information about sensors and the other information about users.

The database of sensors contains three different types of tables:

1. One table with the sensors configuration information.
2. Two tables, one for temperature and the other for humidity, with the information about the alarms for each variable.
3. Many tables as sensors have ever sent information to the application, storing the data sent by that sensor.

The database of users contains two tables:

1. One table containing the information of entry and exit of users to the system.
2. One table with the access data of the users.

Functionalities. The most important part of this project was the development of the application in LabVIEW which has the following functionalities.

User access control. When starting the application it asks the user for a username and a password to be checked with the list stored in the database of users to allow or deny access. Each user has an associated fee: Administrator, manager or operator. Each one gives different permissions on the application, making each functionality accessible or not for that user.

Storage into database of sensor of the values of temperature and humidity sent by the sensors. Each sensor has a table with his name in the database of sensors to store the received values and the date they were taken. In case a sensor doesn't send information when scheduled, the application fills the humidity and temperature columns with a value of 0 indicating they are pad values. An example of data stored into the database can be found in Figure 2.

Humedad	Temperatura	Fecha	Padeo
46,065897504	23,3106	19/09/2009 16:14:31	No
46,032518656	23,3106	19/09/2009 16:14:36	No
45,930393216	23,3008	19/09/2009 16:14:43	No
45,898947264	23,3106	19/09/2009 16:14:47	No
45,863586688	23,3008	19/09/2009 16:14:52	No
45,898947264	23,3106	19/09/2009 16:14:55	No
45,900901776	23,3204	19/09/2009 16:15:02	No
46,1345984	23,3204	19/09/2009 16:15:06	No
46,236632448	23,3302	19/09/2009 16:15:10	No
0	0	19/09/2009 16:15:15	Si
0	0	19/09/2009 16:15:20	Si
46,48485736	23,3204	19/09/2009 16:15:23	No
46,48485736	23,3204	19/09/2009 16:15:27	No
0	0	19/09/2009 16:15:32	Si
46,0050048	23,34	19/09/2009 16:15:36	No
0	0	19/09/2009 16:15:41	Si

Figure 2. Example of data stored into the database of sensors. The columns contain the information about humidity, temperature, date and time, and padding.

Real-time representation of received data. It can be visualized the temperature and humidity from one or more sensors against the date where they were taken. In Figure 3 it can be seen the humidity sent by two sensors: 1043 and 1044.

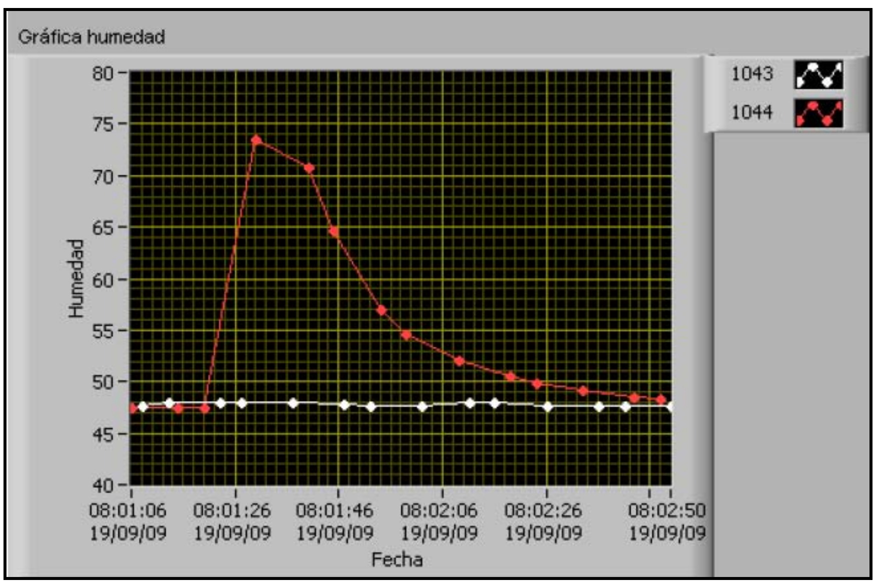

Figure 3. Example of humidity from sensors 1043 and 1044 represented in a real-time graphic.

Intensity diagrams. They represent the surface where the sensors are placed, calculating interpolated information using the information received from the sensors for all the points of the plane. These values are updated every time we received new data from the sensors.

In Figure 4 we can see the intensity diagram with the sensors 1040, 1043 and 1044 placed in the positions (1,0), (2,10) and (4,7) respectively. The values above the names of the sensors show the last value received from that sensor and the background colors show the interpolation value of humidity for the rest of the points of the surface.

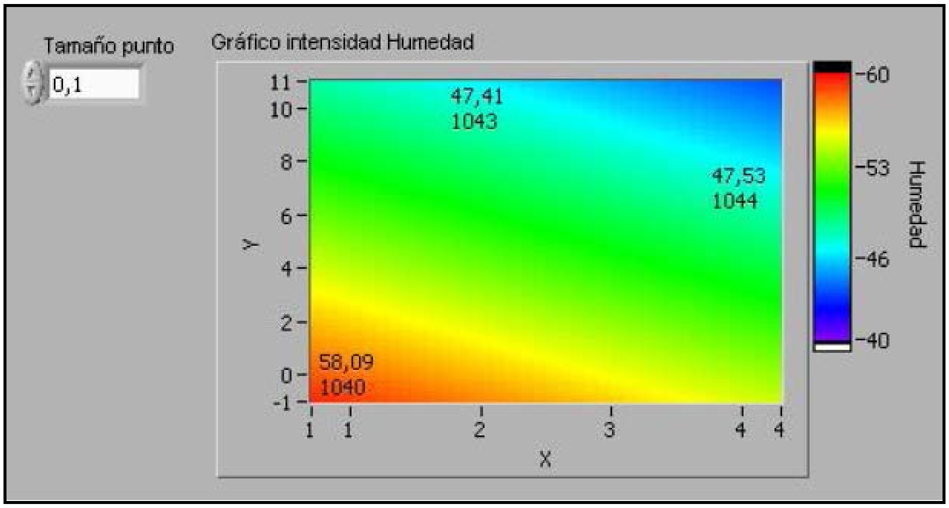

Figure 4. Example of intensity diagram with the sensors 1040, 1043 and 1044.

<u>Control and storage of alarms</u>. produced by values outside the range allowed. Users can set minimum and maximum values for humidity and temperature for each sensor which are stored into the database of sensors. When a sensor returns a value outside of this range an alarm occurs and the indicator light associated to that sensor changes from green to red. This light remains red until a user accepts the associated alarm with a switch, if so, with the next value within the range the light will change to green. The information about the sensor that has had the alarm, the start and finish time, the top or bottom value, the user and the date when it has been accepted are stored into the database. Figure 5 shows a value of humidity of 87.6114 that exceeds 55 which is the maximum value allowed, turning the light indicator Hum. into red. The light indicator Hu. ac. Which is in red indicates that the alarm has not been accepted yet, so the counter on the right indicates there is one alarm waiting to be accepted.

Figure 5. Example of an alarm occurred for sensor 1043.

<u>Visualization in tables and graphics of historical data stored into the database of sensors</u>. A user can see the historical information received for a sensor just choosing the name of the sensor and the ranges of dates to be displayed into a table. Another way to visualize that information can be using a graphic which can be very useful to compare data from several sensors for the same range of time. As a complement of the visualization described above a user can calculate the polynomial approximation of a specific order

that best suits that information. The coefficients of the polynomial calculated are displayed to the user too. The last type of graphical for historical data shows the spline interpolation, which calculates from the data received the most probably values for padding data stored into the database of sensors.

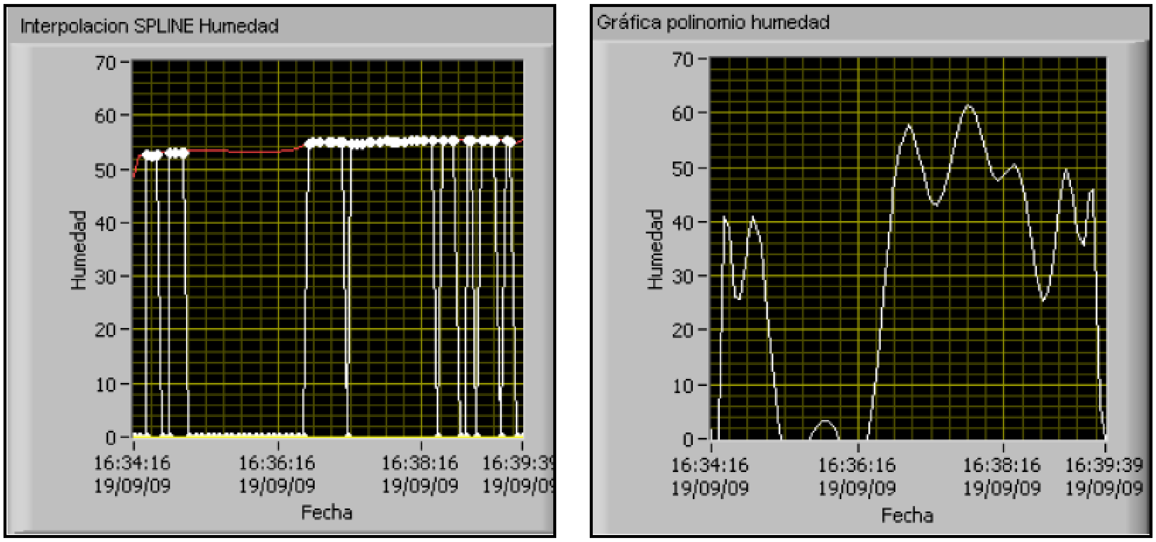

Figure 6. Example of an interpolated graphic of humidity for sensor 1043 (left) and polynomial approximation (right)

<u>Remote access</u>. A user can communicate with the application from another computer with a socket connection.

TESTS AND RESULTS. To check the correct working of the application developed some tests have been made. On one hand we have checked each one of the functionalities implemented to solve concrete problems and on the other we have made tests with an overall working where all the functionalities are working all together. To make these tests we have used the 9 motes MEP510 available.

In this first test 7 of them were working in High Power Mode (HP) which means their radios are always on and the other 2 were working in Low Power Mode (LP) with their radios in sleep mode. The location chosen for this first test was a close chamber of small dimensions, similar to a room of a house. The sensors were placed occupying all the area and their positions were saved into the database of sensors. The acquisition time for all the sensors was 5 seconds, being the theoretical time the sensors last to send new data. In this first test the maximum and minimum values to generate an alarm have been chosen so the probability to generate it is very low. The sensors have been working in 5 sessions of 3 hours, but as the acquisition time is very low the amount of data is high so it can simulate a situation of several hours working with a more common acquisition time of about 1 minute.

The first conclusion of this first test is that the sensors working in HP mode start sending information almost instantaneously, but the sensors with LP mode last a time between 5 and 20 minutes to start sending data. Also the sensors working in HP mode are very regular sending data fulfilling the acquisition time, but the sensors working in LP mode

are very irregular with situations of not sending data for 10 or 15 minutes but then sending several consecutive data.
In this situation there were not alarms and the application worked properly. Then we took one of the sensors producing an increase in humidity. The variation was represented correctly into the real time graphic and the 2D intensity diagram starting an alarm for that sensor. We accepted it and after a few seconds the humidity returned to a normal value. The alarm working was fine and it was registered into the database. This test was repeated with one or more sensors several times and all of them the results were the expected.
The last test made with this configuration consisted of modifying the number of active sensors without stopping the application to check its behavior. First adding a new active sensor to the network and then removing another one from the network. In both cases, when the user changes the number of active sensors into the database, the graphics and the alarms adapted to the new situation, adding or removing that sensor from them.
In the tests made before there were sensors with two different configurations (LP and HP) into the same network. That can be useful to check the difference working between them, but we have also to check a network with all the sensors having the same configuration. In this other case we chose a Low Power configuration as it is the one that allows a higher saving of energy. The other needed change was to modify the location because the previous one was too small to simulate a real future use. The location chosen was a close chamber of 10*6 meters. The 9 sensors were placed in the chamber, but some of them in special placements: one was placed outdoors next to the chamber; two of them were placed inside two cold rooms, one of them at -20 degrees and the other at 5 degrees of temperature; and other one was placed on a heating tube. This way there were high variations of temperature and humidity between sensors and various obstacles between the sensors and the MIB 520 base station such as walls, furniture, windows etc. The results in this case were the expected allowing to check the wide range of temperature and humidity for the sensors. The duration of this test was 3 days so it made it possible to measure the energy consumption for the sensors, confirming the possibility to work a long time with them without charging.

CONCLUSION. This paper shows the development of a SCADA application using LabVIEW 8.6 to handle the information received from a wireless sensors network. The application has the following functionalities:

- User access control.
- Storage into database of the information coming from the sensors.
- Real-time representation into graphics.
- Intensity diagrams.
- Control and storage of alarms.
- Visualization of historical data.
- Remote access.

The application has been tested to check the correct working of the functionalities above. For that, two different environments were used. The first one with a mix of LP and HP sensors in a small room for a few time, and the second with all the sensors working in LP mode in a higher room for a long time and in different climate conditions. In both of them the application worked properly. These tests were also useful to check the advantages: low cost, easy use for temporary locations, good signal to noise relationship and energy saving in multihop networks; and disadvantages: low life-cycle batteries, components with limitations (low speed, memory…) and networks with a huge number of sensors; of the wireless sensors networks.

Acknowledgements. The authors want to thank the means provided by the Department of Agricultural and Forestry Engineering of the University of Valladolid.

REFERENCES

Aakvaag, N., Frey, J. 2006. Redes de sensores inalámbricos, ABB magazine, number 2, pages 39-42.

Crossbow MEP-SYS sensors users manual, revision A. 2007. Available at: https://www.xbow.com/Support/Support_pdf_files/MEP-SYS410_manual.pdf Accessed 9 February 2010.

Hill, J., Szewcyk, R., Woo, A., Culler, D., Hollar, S., Pister, K. 2000. System arquitecture directions for networked sensors, ASPLOS.

Intanagonwiwat, C., Govindan, R., Estrin, D. 2000. Directed diffusion: A scalable and robust communication paradigm for sensor networks. Proc. 6th Annual International Conference on Mobile Computing and Networks (MobiCom 2000), pages 56–67, Boston, ACM Press.

Krishnamachari, B. 2005. Networking Wireless Sensors, Cambridge University Press.

National Instruments Developer Zone. 2009. ¿Qué es una Red de Sensores Inalámbricos? Available at: http://zone.ni.com/devzone/cda/tut/p/id/9028 Accessed 9 February 2010.

Pottie, G.., Kaiser W. 2000. Wireless integrated network sensors, Communications of the ACM.

Yingshu L., My T.T., Weili W. 2008. Wireless Sensor Networks and Applications, Ed. Springer.

Zhao, F., Guibas, L.J. 2004. Wireless Sensor Networks, Morgan Kaufmann.