Computers and Electronics in Agriculture 166 (2019) 105028

Contents lists available at ScienceDirect

Computers and Electronics in Agriculture

journal homepage: www.elsevier.com/locate/compag

Original papers

An IoT-based cognitive monitoring system for early plant disease forecast

Ahmed Khattab[a,*], Serag E.D. Habib[a], Haythem Ismail[b], Sahar Zayan[c], Yasmine Fahmy[a], Mohamed M. Khairy[a]

[a] Electronics and Electrical Communications Engineering Department, Cairo University, Giza, Egypt
[b] Department of Engineering Mathematics, Cairo University, Giza, Egypt
[c] Plant Pathology Research Institute – Agriculture Research Center, Giza, Egypt

ARTICLE INFO

Keywords:
Internet of Things (IoT)
Wireless sensor network (WSN)
Precision agriculture (PA)
Epidemic disease control
Expert systems
Cognitive architectures

ABSTRACT

In this paper, we develop an IoT-based monitoring system for precision agriculture applications such as epidemic disease control. Such an agricultural monitoring system provides environmental monitoring services that maintain the crop growing environment in an optimal status and early predicts the conditions that lead to epidemic disease outbreak. The agricultural monitoring system provides a service to store the environmental and soil information collected from a wireless sensor network installed in the planted area in a database. Furthermore, it allows users to monitor the environmental information about the planted crops in real-time through any Internet-enabled devices. We develop artificial intelligence and prediction algorithms to realize an expert system that allows the system to emulate the decision-making ability of a human expert regarding the diseases and issue warning messages to the users before the outbreak of the disease. Field experiments showed that the proposed system reduces the number of chemical applications, and hence, promotes agriculture products with no (or minimal) chemicals residues and high-quality crops. This platform is designed to be generic enough to be used with multiple plant diseases where the software architecture can handle different plant disease models or other precision agriculture applications.

1. Introduction

Precision Agriculture (PA) has recently become the main trend in global agriculture. PA emerged in the late 1980's with the matching of grid-based sampling of soil chemical properties with newly developed variable-rate application (VRA) equipment for fertilizers (United Nations International Strategy for Disaster Reduction (UNISDR), 2008; Rogers and Tsirkunov, 2011; International Federation of Red Cross and Red Crescent Societies, 2008; Victoria, 2008; Shaw et al., 2008). PA aims at optimizing the production efficiency and uniformity across the field, optimizing the quality of the crops, minimizing the environmental impact, and minimizing the risk both from income and environmental points of view. One of the main applications of PA that is based on environmental auditing is epidemic disease control. Epidemic diseases have severe impacts on the crop production. The key player in epidemic diseases is the climate changes that occur unexpectedly in time and space, which make their impact more severe (Trout et al., 1997; Ogalo et al., 2008). Typically, farmers are not well prepared to react to such diseases which give the diseases time to spread wider and become more destructive (Campbell et al., 2007; HFP Futures Group Making Space for Science - Humanitarian Policy Dialogue, 2011; Intergovernmental Panel on Climate Change (IPCC), 2011). Furthermore, when the crop is frequently affected by the same disease, the farmers tend to increase the doze of the chemical fungicides. Consequently, the level of the chemical residues in the produced crop increases which not only results in harmful environmental effects, but also increases the cost due to the use of large doses of fungicides.

The Internet of Things (IoT) has recently been considered the state-of-the-art in implementing distributed monitoring and control systems in various application areas. In this paper, we build an IoT-based monitoring system that uses Wireless Sensor Network (WSN) technology and is accessible through the Internet for precision agriculture applications such as epidemic disease control. Our IoT-based plant disease management system aims to achieve sustainable agricultural development. This system is generic enough to be used with multiple plant diseases where the software architecture can handle different plant disease models. In addition, the used sensors and developed expert system software are flexible to be used with different plants in the monitored fields or other precision agriculture applications. While our platform is based on a standard wireless communication layer, it

* Corresponding author.
E-mail address: akhattab@ieee.org (A. Khattab).

https://doi.org/10.1016/j.compag.2019.105028
Received 24 June 2019; Received in revised form 23 September 2019; Accepted 28 September 2019

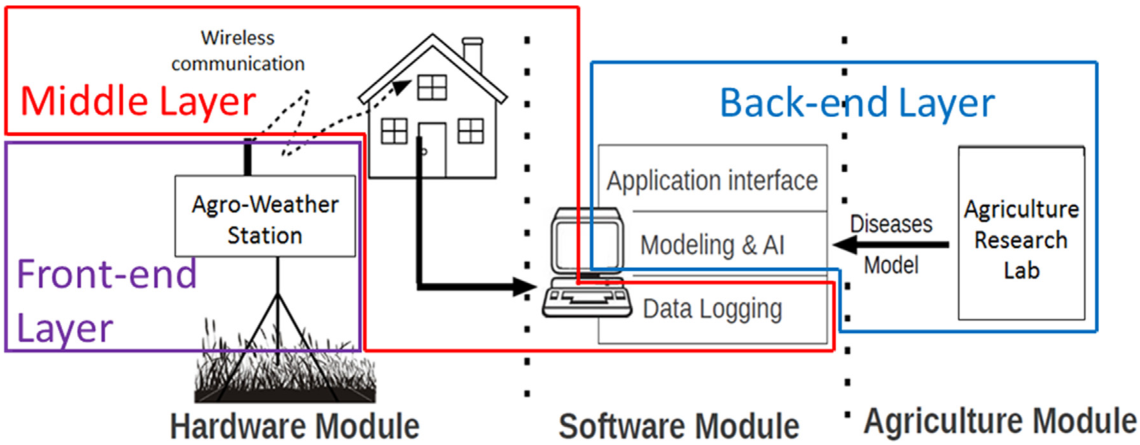


Fig. 1. The proposed monitoring system for early plant disease forecast.

involves a careful system design, since the platform requirements are very strict.

Many research works were dedicated to implement complete platforms for precision agriculture using IoT (see (Jawad et al., 2017) and (Khanna and Kaur, 2019) for detailed literature reviews). Several WSN-based agriculture monitoring and control systems were proposed in the literature for either farms (Hwang et al., 2010; Wenting et al., 2014; Shieh et al., 2011; Navarro-Hellín et al., 2015; Wu et al., 2013; Baggio, 2005) or greenhouses (Liu and Ying, 2003; Zhou et al., 2007; Liu et al., 2007; Ruiz-Garcia et al., 2008; Morais et al., 2008; Manijeh and Amene, 2012). However, no intelligence nor Internet connectivity were implemented in such systems. As precision farming is nowadays popular in the industrialized countries, it is progressing in many other countries and applied research is conducted in countries like India (Gangwar et al., 2019), Brazil (Maia et al., 2017), Uzbekistan (Muzafarov and Eshmuradov, 2019) and Thailand (Muangprathub et al., 2019) among other countries. In (Gangwar et al., 2019), a low-cost agro-ecological resource management system is presented in the Indian context. On the other hand, a field trial in Sao Paulo, Brazil was presented in (Maia et al., 2017) to test a designed real-time, in-situ agricultural IoT device. The device was designed to monitor the soil and the environment. A watering agricultural crops system based on IoT was implemented in Makhamtia District, Suratthani Province, Thailand (Muangprathub et al., 2019). Furthermore, Many IoT systems were suggested to prevent plant epidemic diseases while minimizing cost and environment impact by predicting the required quantity of fungicide to apply based on weather (Foughali et al., 2018; Hadders, 1996, Jaunatre and Gaucher, 2011; Leonard et al., 2001; Spits et al., 2003; Hossam et al., 2018; Ibrahim et al., 2019; Khattab et al., 2016).

The main objective of this work is designing, validating and implementing an IoT-based monitoring system with a custom agro-weather station (hardware and software) optimized for early warning issuance for plant disease epidemics (especially those encountered by major crops in Egypt, where this research is conducted). The contributions of the paper are as follows:

- The hardware design and implementation of an agro-weather station built as part of the integrated monitoring system for plant disease forecasting. The developed agro-weather station is equipped with six sensors to measure the air temperature and humidity, soil temperature and humidity, leaf wetness, rain rate, wind speed and direction and solar radiation intensity. The agro-weather station senses these quantities at a programmable period and sends the sensors' readings as an SMS message to programmable mobile numbers. The agro-weather station is solar-powered and is carefully designed for low power operation. Even though cost optimization was not a design objective, our agro-weather station costs about one third the cost of commercial agro-weather stations.
- The design of a software module that is independent of the types, numbers, and accuracies of the sensors used in the agro-weather station. The developed software provides a reasonable estimate of actual weather conditions, given the sensor data from the agro-weather station and information about the accuracies of the sensors. Furthermore, the software is designed to be independent of any particular disease model. The software module is composed of two components: An Artificial Intelligence (AI) component and a Graphical User Interface (GUI) component. The AI component processes and analyzes the sensors' readings and consequently suggests specific measures that can be taken to protect the crops. Any individual concerned with the generated decisions by the AI module (agricultural engineers, farmers, …, etc.) will access the software module through the GUI web interface.
- We conduct an extensive set of field and laboratory experiments to develop the disease models and then validate the performance of the developed IoT system. We target two crops from the Solanaceae family which are tomato and potato. Our work focuses on three diseases that represent the major threats facing the target crops. These diseases are early blight, late blight and powdery mildew. Each one of these diseases has similar effects on both potato and tomato. This would also facilitate building the software programs that will handle different diseases for both crops.

2. Materials and methods

The proposed monitoring system for early plant disease forecast depicted in Fig. 1 is composed of three main modules: a hardware module, a software module, and an agriculture module. The hardware module is mainly responsible for collecting information from the outdoors environment, then transmitting the collected data using a communication subsystem. The software module has four main functions. (1) It collects, processes, stores, and presents the data provided by the sensors of the different weather stations. (2) It provides a user-friendly interface to the system. (3) It represents and maintains the disease models provided by the agriculture module. (4) It suggests suitable preventive actions (applying pesticides, for instance) by analyzing the collected data in light of the disease models. Finally, the agriculture module is responsible for developing the disease models in different weather conditions based on the collected data. Furthermore, it validates the developed models using extensive field experiments. In this Section, we discuss the research methods used in the hardware and software modules. In the next Section, we elaborate on the experiments conducted within the agriculture module.

The system design follows a layered approach as (Hossam et al., 2018; Ibrahim et al., 2019; Khattab et al., 2016; Yelamarthi et al., 2016). More specifically, the system architecture is composed of 3 layers: a front-end layer, a middle layer, and a back-end layer. The three modules of the system are mapped to the architecture as follows. The front-end layer contains the hardware modules represented by the agro-weather station and the attached sensors and other physical devices that collect information from the outdoors environment. The middle layer supports the communications between the front-end and the back-end layer. Furthermore, the middle layer stores and converts the information collected from the sensors into a database in the back-end layer. Meanwhile, the software module and the agriculture module are implemented in the back-end layer which is a server that resides in our research lab. The back-end layer contains the developed software and artificial intelligence tools that predict different diseases based on the received data and interface the system to the human users.

It is worth mentioning that such a monitoring system is not limited to disease control but can also be used in other precision agriculture applications like pest control and greenhouses.

2.1. System design methodology

First, we discuss the design methodology of the different layers of the proposed system.

2.1.1. Front-end layer design

The front-end layer of our monitoring system is the hardware module, also called the agro-weather station, which is designed to collect the measurements of eleven environmental attributes (using six sensing devices) and send them wirelessly to a remote host (i.e. the back-end) for analysis. The environmental sensors measure the following physical quantities: air Temperature, air humidity, soil temperature, soil volumetric water content, soil electrical conductivity, soil relative dielectric permittivity, wind speed, wind direction, rain level, infra-red and visible solar radiation, and leaf wetness. These physical attributes are needed for the development of the disease models in the agriculture module. The environmental sensors are interfaced to a microcontroller which collects the measurements from sensors and serially sends them to the cellular transceiver (the middle layer), which in turn sends them as a single SMS to the back-end host. The station is solar powered in daytime and battery powered at night. The battery capacity is chosen such that it can power up the station solely for not less than 2 successive days. This is possible because the system is designed to consume very low power. The overall block diagram of the proposed agro-weather station is shown in Fig. 2. The microcontroller also sends status information about the batteries and sensors health.

The specifications of the developed agro-weather station are divided into general specifications and environmental sensors specifications. The general specifications of the agro-weather station cover the power supply, flexibility, mounting and communication method with the back-end (which is a component of the middle layer), while the environmental sensors specifications cover the specifications of the sensors only. The specifications of these environmental sensors are and summarized in Table 1.

The general specifications of the agro-weather station are summarized as follows:

Power Supply: The station is solar powered. The battery is able to supply (in the absence of solar energy) the station for at least two consecutive days.

Flexibility: The station is upgradeable. It should be compatible with sensors of different interfaces such that it would be easy to replace current sensors with higher specification ones in the future.

Mounting: Rugged hardware mounting is required (e.g. galvanized steel tower).

2.1.2. Middle layer design

The middle layer wirelessly transfers the collected sensors' data to the back-end layer where it will be stored and processed for decision making. Wireless transmission reduces and simplifies wiring, allows deploying the sensors at remote, dangerous, and hazardous location, easy installation, and integration for extremely low cost, small size and low power requirement and mobility. Different wireless communication standards can be used for our system such as Cellular Communications (GSM/GRPS, 3G or 4G), IEEE 802.11 (WiFi), IEEE 802.15.4 (ZigBee), and narrow-band IoT (NB-IoT) which are widely used for measurement and automation applications. Since our system spans a wide agricultural area, we use cellular services due to its relatively long-range wireless communication and its robust communication links. More specifically, the communication between the agro-weather station and the back-end host is accomplished using a SIM card. The agro-weather station sends the sensors' measurements and system status information (such as the batteries and sensors health). It also receives commands (e.g., control and reset) or controlling actions from the back-end host, and accordingly takes the appropriate action(s) such as issuing a warning. A cellular transceiver module is interfaced to the micro-controller of the agro-weather station to achieve such a bi-directional communication link.

Another important component of the middle layer is data logging. The data logging component is the interface between the software modules implemented on the back-end layer and front-end layer represented by the agro-weather station. Since the agro-weather station sends the collected sensors' data wirelessly to a file, the data logging module only needs to parse that file and extract relevant data.

2.1.3. Back-end layer design

One of the targets of our work is to develop an *expert system* which is an intelligent software system that is capable of automatically suggesting whether the farmer should apply fungicide or not, based on information from the agro-weather station. The system should also be flexible and scalable to allow for adding new models or new theories captured from real-life experience. The ability to easily add more models in the future for more diseases and customizing the system is emphasized.

The back-end layer hosts the software module that is composed of software tools that implement the expert system and make it accessible to authorized humans connected to the Internet. The software design process is demarcated by the four functions listed at the beginning of this Section and guided by the following principles:

P1. The design of the software is to be independent of the types, numbers, and accuracies of the sensors used in the agro-weather station.

P2. The software is to provide a reasonable estimate of actual weather conditions, given sensor data from the agro-weather station and information about the accuracies of the sensors.

P3. The software design is to be independent of any particular disease model.

Fig. 1 depicts the main components in the design of the software module. It operates on the data logged by the middle layer. The application interface is a graphical user-interface (GUI), through which users may interact with the system. The GUI allows the users to inspect sensor data, set system parameters describing the sensors connected to the system, and inspect and modify the disease models.

The artificial intelligence (AI) module is the heart of the expert system. It is made up of a GLAIR (Grounded Layered Architecture with Integrated Reasoning) agent. The GLAIR agent is an intelligent agent based on the GLAIR cognitive architecture (Shapiro and Ismail, 2003; Shapiro and Bona, 2010). This agent is responsible for collecting the sensors' data from the agro-weather station (through the data logger), fusing the data over time and space to achieve a reasonable estimate of the actual weather conditions, and representing said information as statements of a formal language over which logical reasoning can take place. With a representation of the disease model in the same formal

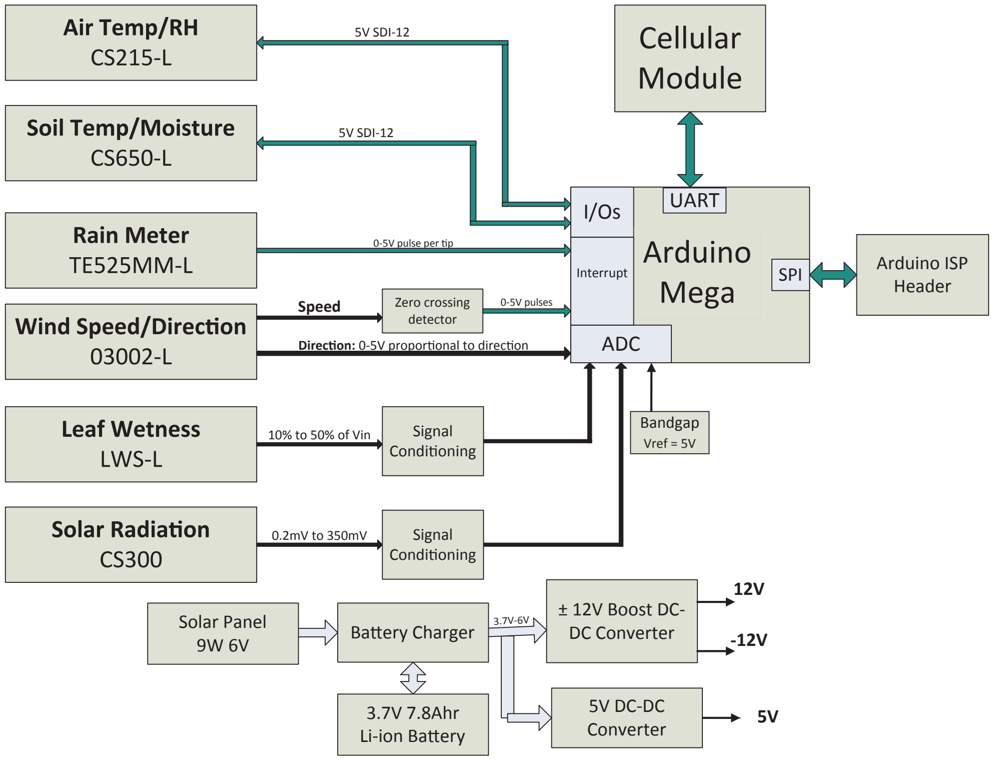


Fig. 2. Agro-weather station detailed block diagram.

Table 1
Environmental sensors specifications.

Sensor	Range	Accuracy	Resolution
Air Temperature	−10° to 50°	± 1°	1°
Air Humidity	0 – 100%	± 2%	0.5%
Soil Temp.	−15° – 60°	± 2°	1°
Soil Vol. Water Content	5 – 50%	± 3%	1%
Electrical Conductivity	0 – 8 dS/m	± 5%	0.5%
Relative Permittivity	1 – 81	± 3%	< 0.02
Wind Speed	5 – 100 Km/h	± 2 Km/h	2 Km/h
Wind Direction	0° – 360°	± 3°	22.5°
Rain Meter	–	–	0.2 mm
Solar Radiation (IR + Visible + UV)	360 nm to 1120 nm Max Irradiance: 1750 W/m^2	± 5%	1 W/m^2
Leaf Wetness Sensor	0 – 100%	–	–

language, the agent reasons about preventive actions to be taken should the need arises.

We claim that our design fairly meets the three design principles **P1**, **P2**, and **P3** listed above. **P1** is met by the generic implementation of the GLAIR agent which only assumes the existence of different sensor types, with each type instantiated by some number of actual sensors each with some accuracy. Through the GUI, users may *define* different assortments of sensors (stations) which the agent can maintain and reason about separately. **P2** is observed by the GLAIR agent's employing sensor data fusion techniques (Khaleghi et al., 2013) to arrive at relatively accurate information about weather conditions given noisy sensor data transferred through possibly lossy wireless channels. **P3** is met by choosing to represent all information and disease models declaratively as a set of logical sentences in the knowledge base of the GLAIR agent. This choice allows us to painlessly revise the model if needed using the GUI (Alchourrón et al., 1985). In what follows, we explain the GLAIR cognitive architecture in detail.

GLAIR (Grounded Layered Architecture with Integrated Reasoning) is a multi-layered cognitive architecture for embodied agents operating in real, virtual, or simulated environments containing other agents (Shapiro and Ismail, 2003; Shapiro and Bona, 2010). The highest layer of the GLAIR architecture is called the *Knowledge Layer* (KL). The KL contains the beliefs of the agent, and is responsible for conscious reasoning, planning, and acting. At the other end, the lowest layer of the architecture is the *Sensori-Actuator Layer* (SAL). The SAL contains all the controllers for the sensors and actuators of the hardware robot. Between KL and the SAL, there is an intermediate layer: the *Percepto-Motor Layer* (PML). The PML is responsible for all the necessary communication between the KL and the SAL (see Fig. 3).

The KL is the layer where all conscious reasoning, planning and action selection takes place. It contains all the beliefs of the agent including short– and long–term memory, quantified and conditional beliefs used in the reasoning process, plans for executing complex actions and achieving goals, beliefs about the preconditions and effects of the different actions, and policies about conditions under which actions should be performed. The KL is implemented using the Semantic Network Processing System (SNePS) for representation, reasoning and acting (Shapiro and Group, 2010).

The PML can be thought of as three sub-layers: (1) **PML$_a$** which is responsible for grounding the KL symbols into perceptual structures and subconscious actions. It also provides the agent with a sense of situatedness (such as the sense of "I", "You", and "Now") by maintaining a set of deictic registers; (2) **PML$_b$** that handles the communication between PML$_a$ and PML$_c$; and (3) **PML$_c$** that is responsible for abstracting the hardware sensors and actuators into the basic behaviors of the agent.

The three layers provide mind-body modularity to a GLAIR agent. The KL can be thought of as the mind of the agent, while the PML and the SAL layers can be thought of as the body of the agent. The KL and PMLa layers are generally independent of the body implementation and can be connected without modification to any hardware or simulated body.

The Behavior Cycle: A GLAIR agent is either thinking about a percept or answering some question. The acting module has been added to a GLAIR agent to perform actions based on the underlying beliefs of the agent. GLAIR agents essentially execute a sense-reason-act cycle,

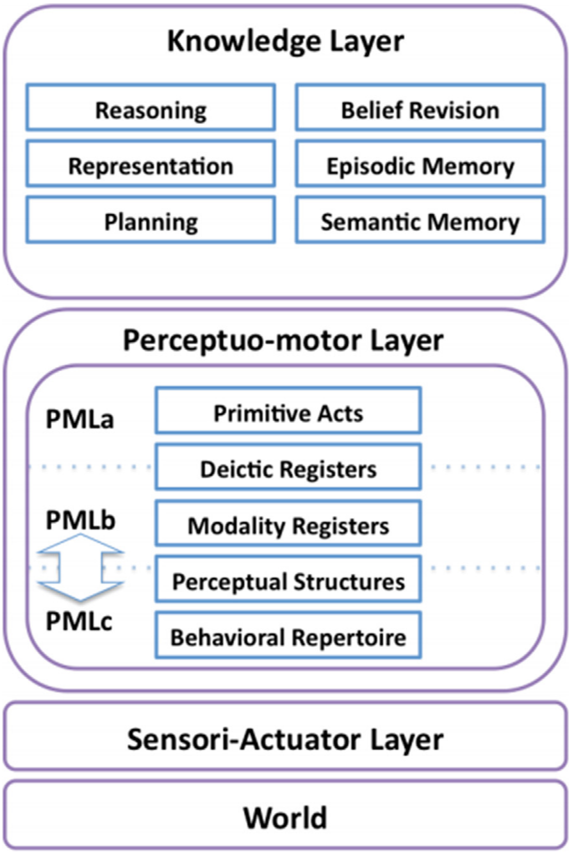


Fig. 3. The GLAIR Cognitive Architecture.

but not necessarily in this strict order. The basic behavior cycle of a GLAIR agent consists of four main steps:

1. Take an input from the external environment. This input can be a natural language utterance, an input statement, query, or command.
2. Analyze the input. This process might trigger reasoning causing new beliefs to be added to the KL.
3. If the input is a statement, then the input proposition is added to the KL. This operation might trigger belief revision if the input contradicts the current belief state of the agent. If the input is a query, then backward inference is triggered, and the result of the inference will be returned. If the input is a command, then the desired act is performed, and a confirmation proposition is output indicating that the agent has successfully performed the act.
4. If the input was a natural language utterance, an output utterance representing the output proposition(s) is generated.

2.2. System implementation

Here, we detail our implementation of the different layers of the proposed system.

2.2.1. Agro-weather station hardware implementation

In this section, we present the implementation details and the components selected for our agro-weather station including the sensors, microcontroller and power supply. We also present the overall integrated system design.

Sensors: We carried out an in-depth survey of the available off-the-shelf components in the market. Based on the required specifications of our agro-weather station listed in Table 1, we selected the sensors shown in Fig. 2 that achieve our target technical specifications.

Microcontroller: For our agro-weather station design, we used the Microchip ATmega2560 microcontroller with an open source Arduino bootloader (Microchip Technology, 2019). The ATmega2560 is an 8-bit AVR RISC-based microcontroller that combines 256 KB ISP flash memory with read-while-write capabilities, 4 KB EEPROM, 8 KB SRAM, 16-channel 10-bit Analog-to-Digital converter (ADC), wide variety of I/O peripherals, and six software selectable power saving modes. The Arduino bootloader enables us to easily interface the ATmega2560 to the environmental sensors and to the cellular module by providing an integrated development environment with a variety of open source libraries (Microchip Technology, 2019).

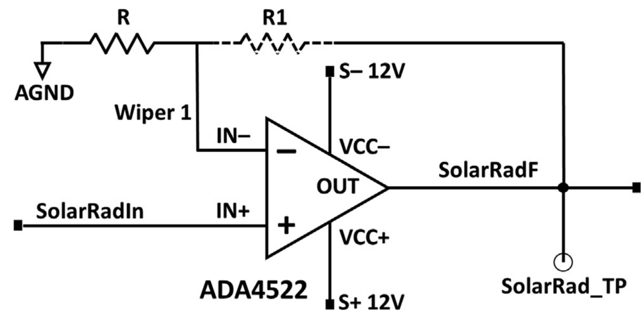


Fig. 4. Design of the analog conditioning circuit connected to the solar radiation sensor. R1 is implemented via the digitally programmable potentiometer AD5204 from Analog devices.

Signal Conditioning Circuits: They are needed to adapt the analog sensors outputs to the built-in Arduino ADCs. For example, consider our interface circuit of the solar radiation sensor to the Arduino built-in ADC depicted in Fig. 4. The ADA4522-1 op-Amp (Analog Devices, 2019) amplifies the sensor output to match it to the Arduino ADC input range. The AD5204 digitally-controlled potentiometer is used to enable precise control of the gain of this Op-Amp. We omit the details and performance evaluation of such a circuit and other signal conditioning circuits to not disrupt the readability of the paper.

Power Supply: The worst-case power consumption of our system is only 0.5 Watthours per day. It needs three power supply rails; 5 V, +12 V and −12 V. The 5 V rail is used by the microcontroller and the cellular communication module. The ± 12 V are used by the sensors and the signal conditioning circuits.

Battery: We use a 7000 mAh 3.7 V polymer lithium ion battery. This battery allows for 50% depth of discharge. It supplies the station with the needed power for not less than 2 consecutive days without being charged.

Solar Panel: A 6 V 9 W monocrystalline solar panel is used to power the weather station as well as to charge the battery (Voltaic Systems, 2019).

Battery Charger: Microchip MCP73871 battery charger IC is selected. It is a fully integrated linear solution for system load sharing and Li-Ion/Li-Polymer battery charge management with ac-dc wall adapter and USB port power sources selection. It is also capable of autonomous power source selection between input and battery.

Power Management System: Low power consumption is a very important design requirement for our agro-weather station. If this requirement is not strongly pursued, the battery size, solar panel size, and system size and weight will be impacted adversely. The agro-weather station sends messages quite infrequently. The time to acquire the sensors readings and wirelessly transmit them is 2 s. This process is repeated once every 10 min. To maintain low power consumption, we turn most of the system completely off between measurements, rather than turning them to sleep mode. Only the system parts that are necessary for system wakeup at the start of the upcoming measurements are kept active.

Fig. 5 shows our agro-weather station (with a closeup view of its controller box) deployed in an agriculture field in the Agriculture Research Center in Giza, Egypt.

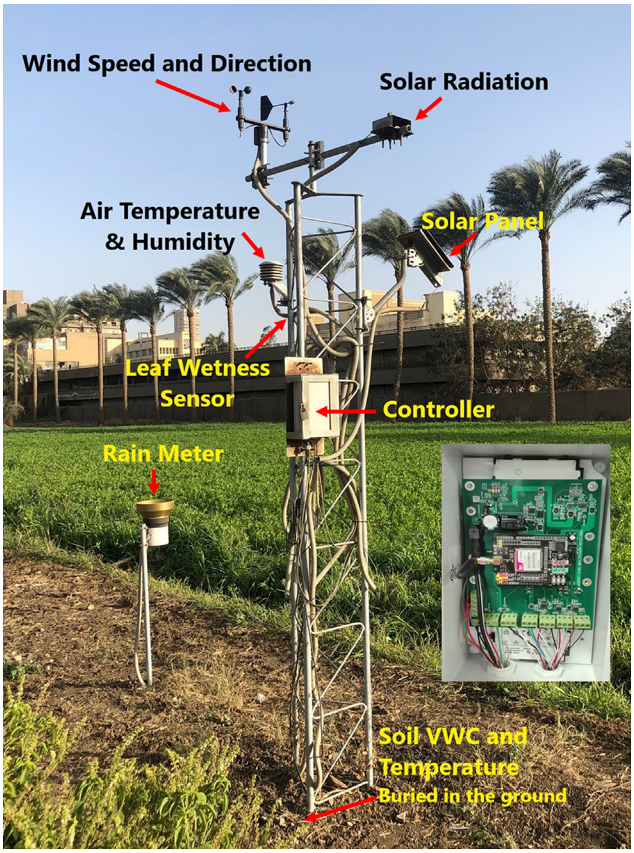


Fig. 5. The agro-weather station installed in an agriculture field.

2.2.2. Middle layer implementation

The main component of the middle layer is the wireless communications mechanism. The SIM900 GSM transceiver from LinkSprite is selected for establishing the communication between the station and the host. The SIM900 transceiver supports GSM/GPRS 850/900/1800/1900 MHz voice, SMS, Data and Fax in a small form factor and with low power consumption (LinkSprite, 2019). It is configured and controlled via its UART using simple AT commands.

Fig. 6 shows an image of an integrated SMS message sent from our agro-weather station to the back-end host alongside the abbreviations used for the reading of each sensor.

2.2.3. Software implementation

Fig. 7 shows the structure of the AI module. Our implementation has a complete PML, including an implementation of Bayesian multi-sensor data fusion (Durrant-Whyte and Henderson, 2008). Through a simple *configuration file*, the PML gets specifications of one or more agro-weather stations to which the system is to be connected. (Think of these as multiple SALs.) Parameters of the configuration file are to be set through the graphical user interface, and include types and numbers of sensors, relevant ranges of sensor readings, and the duration of the time interval separating two consecutive readings. Readings are fused over time and space, and statements, annotated with confidence degrees, are sent to the KL indicating likelihoods of different possible weather conditions (for each sensor type).

The KL component is also implemented. Receiving uncertain data from the PML (likelihoods of different estimations of various weather conditions), however, required an extensive revision of the SNePS system to allow reasoning under uncertainty. The semantics of the resulting reasoning system has been published in (Ismail and Ehab, 2015).[1] The need for reasoning under uncertainty may be illustrated by the following simple example.

Illustrative Example: Consider a GLAIR agent reasoning about the temperature in the environment. If the temperature is high, then a disease is more likely, and a warning should be generated. The following represent the initial beliefs of the agent:

1. The temperature is always exactly one of *low* or *high* or *unknown*.
2. If the temperature is *high*, generate a warning.

The PML fuses the raw readings from the temperature sensors, and feed the KL with new beliefs as follows:

3. The temperature is *high* with a confidence level of 0.692 at t_1.
4. The temperature is *low* with a confidence level of 0.154 at t_1.
5. The temperature is *unknown* with a confidence level of 0.154 at t_1.

From (3), (4), and (5), the modified SNePS reasoning system derives the following (Ismail and Ehab, 2015):

6. The temperature is *high* at t_1.
7. The temperature is *low* at t_1.
8. The temperature is *unknown* at t_1.

Together with (1), these sentences are contradictory. To resolve the inconsistency, the agent will have to disbelieve propositions (7) and (8) as they have less confidence levels than (6). A warning will be generated since the agent ends up believing that the temperature is high.

The Plant Disease Forecast Agent: The plant disease forecast agent is a GLAIR agent which is responsible for collecting the sensors' data from the agro-weather station (through the data logger), fusing the data over time and space using Bayes theorem to achieve a reasonable estimate of the actual weather conditions (Khaleghi et al., 2013), and representing said information as graded statements of a formal language over which logical reasoning can take place. Since the mind of the GLAIR agent is implemented in GSNePS, all the statements will be represented in SNePSLOG (Shapiro and Group, 2010). By representing the disease model in SNePSLOG too, the agent can reason about preventive actions to be taken should the need arise.

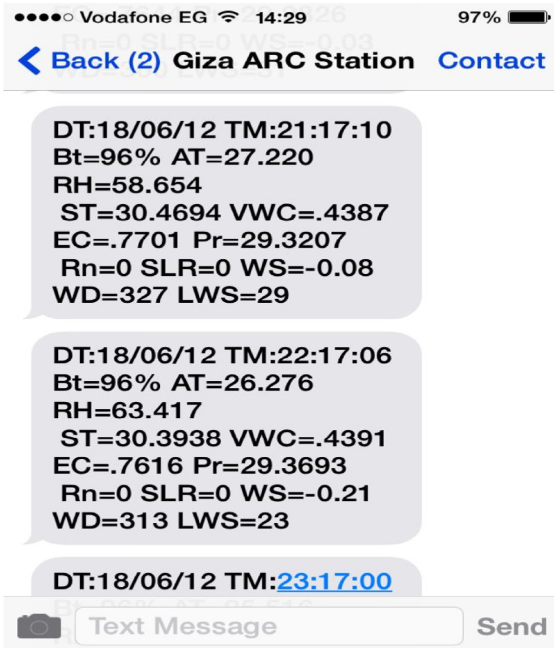
The knowledge level of the GLAIR agent is responsible for conscious reasoning and action selection. A disease model supplied from the agriculture module is represented in SNePSLOG and is loaded to the GLAIR agent as part of the KL once it is selected via the GUI. The SNePSLOG file containing the disease model is itself generated by the GUI as will be detailed below. Fig. 8 shows an example of a SNePSLOG representation of the disease model developed in (Afifi and Zayan, 2009) which is summarized in Fig. 9. Event A is an event which happens if one of three conditions occur:

(i) The temperature remains low and the RH high for six hours (Line 6 in Fig. 8).
(ii) The temperature remains moderate and the RH high for eight hours (Line 7 in Fig. 8).
(iii) The temperature remains high and the RH high for ten hours (Line 8 in Fig. 8).

Likewise, "Event B" is defined as the occurrence of one of three weather conditions (Lines 10 through 15 of Fig. 8). According to (Afifi and Zayan, 2009), a treatment should be recommended if Event B happens within twenty-four hours of an occurrence of Event A, provided that it is not the harvest season and that treatment is not already underway (see Fig. 9).

Graphical User Interface (GUI): The graphical user interface is implemented as a web application. The web interface application is

[1] Interested readers are referred to this paper which provides the details of our reasoning systems.



(a) Actual SMS sample

Abbrev.	Meaning
DT	Date YY/MM/DD
TM	Time HH:MM:SS
AT	Air temp. in °C
RH	Air humidity %
ST	Soil Temp. in °C
VWC	Soil Volumetric Water Content (m3/m3)
EC	Soil Electrical Conductivity (dS/m)
Pr	Soil Dielectric Permittivity
Rn	Rain level
SLR	Solar Radiation (W/m2)
WS	Wind Speed (m/s)
WD	Wind Direction (degrees)
LWS	Leaf wetness

(b) SMS Measurements Abbreviations

Fig. 6. Example of the SMS sent by the agro-weather station to the data logger.

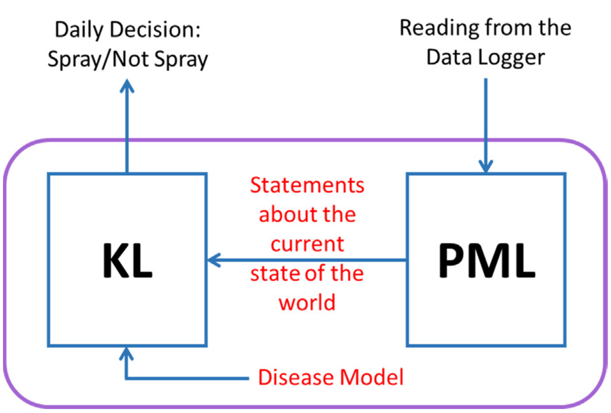


Fig. 7. Structure of the AI Module.

structured into four main components/web pages: home page, disease model definition web page, weather station definition web page, and statistics and graphs web page. Fig. 10 depicts a sample disease model definition page of the developed GUI. A dropdown list allows the user to specify the number of detected events (up to 4). The web page can have as many panels as needed for the detected events. On each panel, the user can choose a phenomenon among the following four: temperature, relative humidity, leaf wetness, and global radiation; in addition to a range and duration for each phenomenon. These phenomena were specifically chosen according to the availability of the sensors in the developed hardware.

3. Results and discussion

In our experimental evaluation of the developed system for early plant disease forecast, we consider two important crops in Egypt: tomato and potato. We consider the most common diseases that affect such crops such as early blight (in potato and tomato), late blight (in potato and tomato), and powdery mildew (in tomato). First, we use the developed agro-weather station to collect the environmental data needed by the agriculture module to develop the different disease models from ten different Egyptian governorates. Then, we implement the developed models in the software module to replace the models in (Afifi and Zayan, 2009) that were used during the initial software development phase (depicted in Figs. 8 and 9). Finally, we validate the disease models through field trials in two governorates.

3.1. Data collection for disease modelling

Samples of the different considered diseases were obtained from naturally infected tomato and potato plants. These samples were collected from ten different Egyptian governorates (Bani-Swef, Beheira, Faiyum, Gharbia, Giza, Ismailiya, Kafrelsheikh, Menoufya, Qalyubia and Sharqia) during different growing seasons. The infected fresh leaves and fruits (tomato) were transferred from the field to the laboratory in transparent polyethylene bags. The obtained samples were rinsed and washed with tab water to remove dust and incubated in humid chamber at 17 °C for three days at darkness for enhancement of sporulation. The diseases were isolated from these samples and used in both field and growth chamber environments to correlate the disease development with the environmental attributes. The growth chamber was used to provide uniform, favorable, repeatable environmental conditions and permit several cycles of screening per year, thus offering more reliable results. Growth chamber and field test results were found to correspond well.

The experimental design used in the current study was a randomized complete block design with three replications. Each replica consisted of three plots. The first plot was treated by recommended period fungicides, the second plot was according to forecasting and early warning information and the last plot was without fungicides for control (check) purposes. Each plot was one ridge of 6 m in length and 1.25 m in width, and hence, the plot area was 7.5 m^2. The distance between plants was 50 cm, and each plot contained 12 plants (one plant per hill). Cultural practices, such as fertilization, irrigation, and weed and insect control were performed whenever they were necessary, as recommended for commercial potato and tomato production. Plant diseases were recorded daily (percent blighted foliage per plot) throughout the 2016 and 2017 growing seasons.

Disease severity was recorded on 100 leaves randomly selected from

```
all(t)(andor(1,1){Holds(temp(low),t), Holds(temp(mod),t), Holds(temp(high),t)}).
all(t)(andor(1,1){Holds(rh(low),t), Holds(rh(high),t)}).
all(t)(andor(1,1){Holds(ppt(low),t), Holds(ppt(high),t)}).
all(t)(andor(1,1){Holds(wetness(low),t), Holds(wetness(high),t)}).

all(t)(duration(temp(low),6,t) => (duration(rh(high),6,t) => EventA(t,temp(low)))).
all(t)(duration(temp(mod),8,t) => (duration(rh(high),8,t) => EventA(t,temp(mod)))).
all(t)(duration(temp(high),10,t) => (duration(rh(high),10,t) => EventA(t,temp(high)))).

all(t)(duration(temp(low),6,t) => (duration(rh(high),6,t) => (duration(ppt(high),6,t)=>
      (duration(wetness(high),6,t)=>EventB(t,temp(low)))))).
all(t)(duration(temp(mod),8,t) => (duration(rh(high),8,t) => (duration(ppt(high),8,t)=>
      (duration(wetness(high),8,t)=>EventB(t,temp(mod)))))).
all(t)(duration(temp(high),10,t) => (duration(rh(high),10,t) => (duration(ppt(high),10,t)=>
      (duration(wetness(high),10,t)=>EventB(t,temp(high)))))).


all(x,t1)(EventA(t1,x) => (all(t2)(EventB(t2,x) => whendo(EventA(t1,x),ck-dur(t1,t2,24))))).

all(t1,t2)(~Harvest(t2) => (~Active(t2)=> (hours(t1,t2) =>(all(x) (EventA(t1,x) =>
    whendo(EventB(t2,x), spray(t2))))))).
```

Fig. 8. A disease model in SNePSLOG.

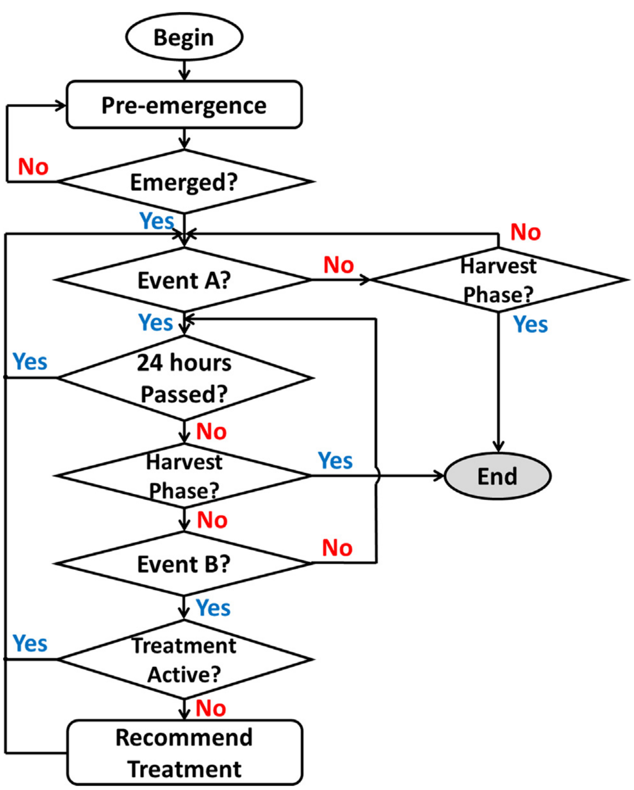


Fig. 9. The disease model of Late Blight according to Egy-blightcast (Afifi and Zayan, 2009).

each replication and using the 0–5 rating scale depicted in Table 2. The disease severity index (DSI) was calculated using the following formula

$$\text{Disease Severity Index (DSI)} = \frac{\text{Sum of all individual ratings}}{\text{No. of examined leaves} \times \text{Maximum disease scale}} \times 100 \quad (1)$$

The field experiments were conducted during 2015, 2016 and 2017 growing seasons. Fig. 11 depicts a sample of the recorded DSI of both crops in the different fields undertest.

Weekly weather parameters such as the maximum/minimum temperature (°C), morning/evening relative humidity (%), rainfall (mm), rainy days sunshine hours (hr) and wind speed (km/hr) were studied and recorded. Fig. 12 shows a sample of the collected data collected from Beheira governorate. They were correlated to the weekly disease severity index by calculating the Karl person's correlation coefficient (r). The correlation coefficient values were tested individually for their significance at 5% and 1% probability level to determine the effect of the weather factors on the disease development. The correlation coefficient r measures the disease development in terms of the apparent infection rate (unit/day) which is calculated as:

$$r = \frac{2.3}{t_2 - t_1} \times \log\frac{x_2(1 - x_1)}{x_1(1 - x_2)} \quad (2)$$

where, r is the apparent infection rate (units/day), and x_1 and x_2 are the disease severity indices recorded at times t_1 and t_2, respectively.

3.2. Disease model development

One of our goals is to design, evaluate and validate 5 disease models for two crops. Then, we use the AI disease forecasting models developed in the software module to achieve optimum disease control with minimum fungicide use by forecasting the date of disease outbreak and to determine the correct time to begin the fungicide application.

The models were designed based on the collected observations in lab analyzing the correlation between the input variables of environmental factors such as: temperature, relative humidity, leaf wetness, precipitation global radiation, wind speed and its effect on the disease incidence and severity of disease causal agents the pathogens, to be calculated. A consequent recommendation in the form of a Spray/Don't Spray message to guide the fungicides application for perfect disease control in the appropriate time. Finally, the model recommendations were compared to disease management by traditional routine spray schedules. We used the developed agro-weather station to monitor the microclimate conditions. The models were tested several times under laboratory conditions before switching to the validation using field experiments conducted at Beheira and Fayoum governorate throughout

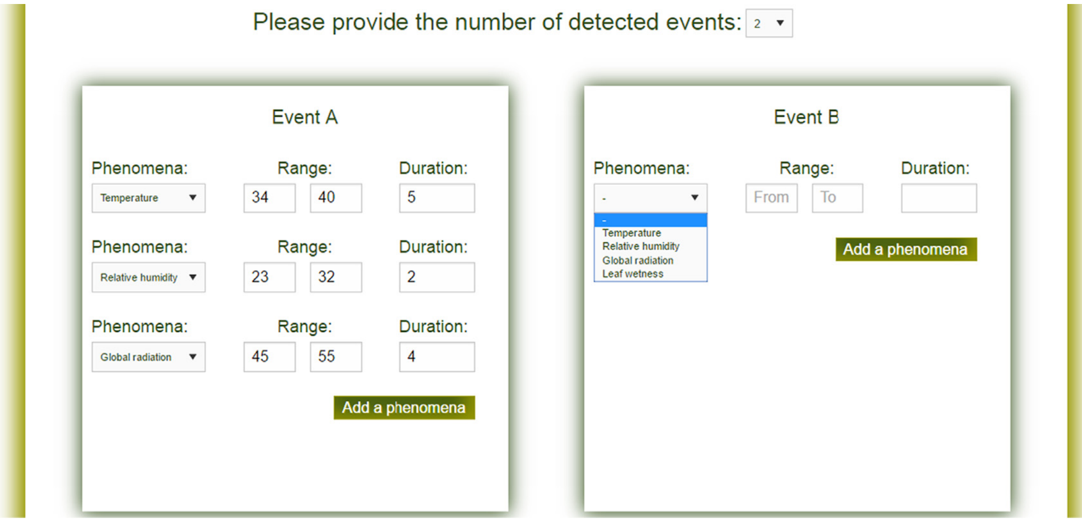


Fig. 10. Example disease definition web page with 2 weather events as the case with the considered diseases.

Table 2
Disease rating scale.

Rating	Disease Condition
0	Free from infection
1	One or two necrotic spots on a few lower leaves of plant
2	A few isolated spots on leaves, covering nearly 5%–10% of the surface area of the plant
3	Many spots coalesced on the leaves, covering 25% of the surface area of the plant
4	Irregular, blighted leaves and sunken lesions with prominent concentric rings on the stem, petiole, and fruit, covering 40%–50% of the surface area
5	Whole plant infected; leaves and fruits starting to fall

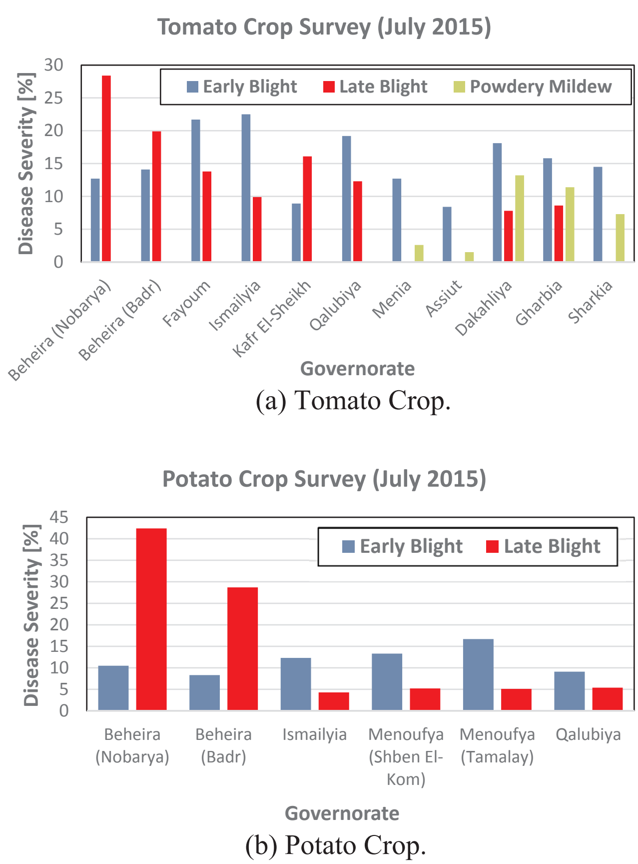


Fig. 11. Sample of the recorded DSI for the month of July 2015.

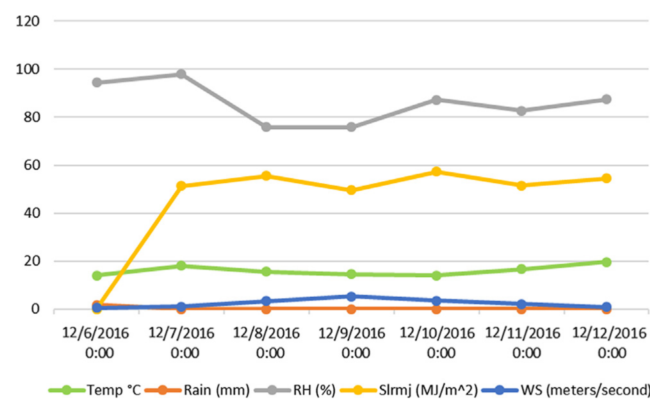
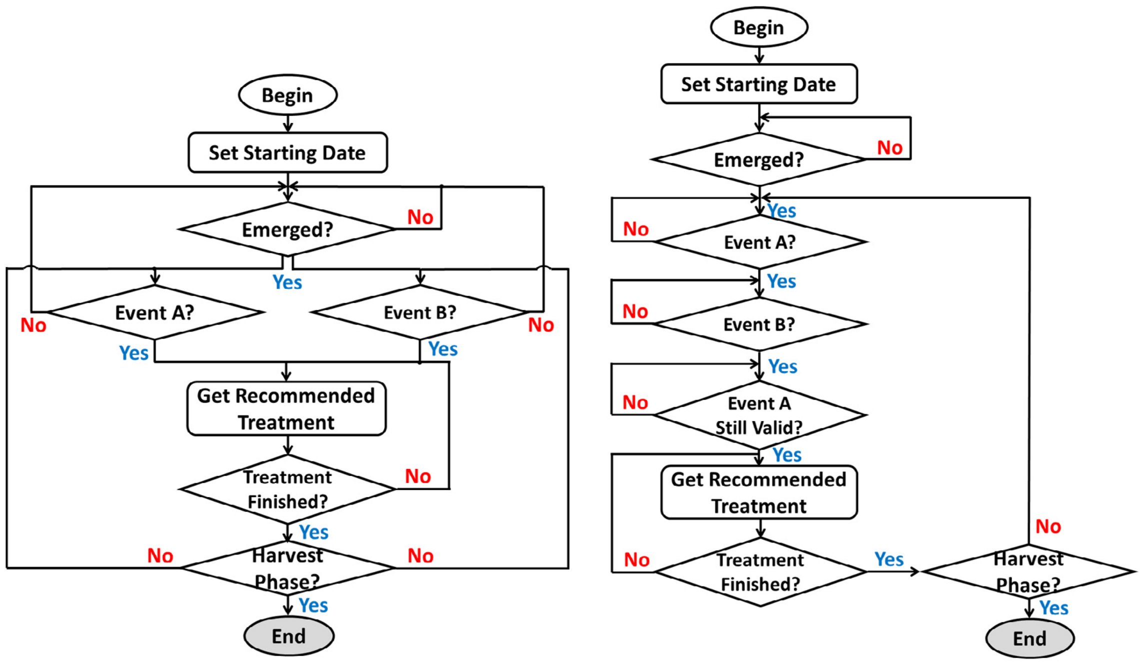


Fig. 12. Sample of the recorded environmental data in Beheira governorate.

two potato growing seasons. Then, the model was revised and refined based on the collected findings.

In what follows, we explain the details of each designed model for each disease. More details of X, Y, and Z as well as the definition of events A and B are available in (Afifi and Zayan, 2009).

Early blight caused by Alternaria solani: The model evaluation and validation follow the basic rules of system analysis to identify events (A and B) for spore germination according to the relative humidity (RH) and leaf wetness (LW), respectively, as presented in Fig. 13(a). The model is activated at the pre-emergence phase. It tries to detect events A and B which are defined as: Event A is triggered when the model detects at least X accumulated dynamic summation hours of RH > 90% and temperature between Y and Z according to the data tabulated in Table 3. While event B is triggered when the model detects at least X accumulated dynamic summation hours leaf wetness > 2.5 U and/or precipitation > 0.2 mm/hour and temperature between Y and Z according to the data tabulated in Table 4.



(a) Potato early blight disease model **(b) Potato late blight and tomato powdery mildew model**

Fig. 13. Flowcharts of the forecasting models developed for potato and tomato diseases under test.

Late blight caused by Phytophthora infestans: Since the pathogen of late blight is the same for potato and tomato, we designed a model for Phytophthora infestans. The only difference in application is to consider the age if susceptibility for each crop. The model implementation follows the basic rules of system analysis to identify events (A and B) according to Fig. 13(b). The model starts after the emergence phase is activated, and it tries to detect the event A, which is defined as: Event A is triggered when the model detects at least X accumulated dynamic summation hours of RH > 90 and temperature between Y and Z according to the data tabulated in Table 5. Then the model looks for event B which is continues hours of RH > 90, and leaf wetness > 2.5 U, and/or precipitation > 0.1 mm/hour. The B event must happen through the 24 h of the A event, but not late than 24 h after an A event. Moreover, using the model system analysis brings an extension to the model, which applies the same rules not only to identify the first critical phase (DIP) of the season, but also to issue warnings throughout the whole growing season.

Tomato Powdery Mildew caused by Leveillula taurica (Ĺev.): Fig. 13(b) presents a flowchart of the model which follows the basic rules of system analysis to identify events (A and B). Similar to the model of the potato late blight, the model starts after the emergence phase is activated, and it tries to detect the event A, which is defined as: Event A is triggered when the model detects at least X accumulated dynamic summation hours of RH > 60 and temperature between Y and Z according to the data tabulated in Table 6. Then the model looks for event B which is continues hours of RH > 60. We omit further details for space limitation.

3.3. Model implementation and field testing

Field experiments with the developed models implemented in the software module were conducted in big plots at Beheira and Fayoum governorates throughout potato and tomato growing seasons in 2017 and 2018. All the selected cultivars are susceptible to the considered diseases. Such experiments were designed as randomized complete blocks and the area of each plot was 12 m^2 (3 m × 4 m). Four replicas were used for each treatment and non-treated plots (sprayed with only water) served for control purposes.

The following three treatments were tested: (i) a full-schedule fungicide program, in which plants were sprayed every 10 days; (ii) a full-schedule fungicide program, in which plants were sprayed every 7 days; (iii) spraying when nominal use of the developed expert system software indicated that a spray application was needed and at least 10 days had elapsed since the previous fungicide application. Weekly inspections of potato and tomato fields were initiated to ensure clear detection of the early sign of diseases. Weather data were automatically collected 24 h a day by the developed agro-weather station deployed within the canopy of potato and tomato field. The data presented in Fig. 14 present the collected weather data for Fayoum and Beheira governorates for 2017 and 2018 seasons during the model validation experiments. Disease severity was measured 15 days after the last spray using a randomized sample of one hundred leaves from every plot, control efficacy of both treatments was recorded.

The results of such investigation demonstrated that the developed forecast models for the considered potato and tomato diseases are

Table 3
The basic rules of system analysis to identify event A in the potato and tomato early blight forecasting model.

Y = Temp. From:	Z = Temp. To:	X = 4 h of RH	X = 6 h of RH	X = 8 h of RH
27 °C	31 °C	Event **A** detected		
23 °C	26 °C		Event **A** detected	
18 °C	22 °C			Event **A** detected

Table 4
The basic rules of system analysis to identify event B in the potato and tomato early blight forecasting model.

Y = Temp. From:	Z = Temp. To:	X = 2 h of L W	X = 4 h of LW	X = 6 h of LW
27 °C	31 °C	Event **B** detected		
23 °C	26 °C		Event **B** detected	
18 °C	22 °C			Event **B** detected

Table 5
The basic rules of system analysis to identify event A in the potato and tomato late blight forecasting model.

Y = Temp from:	Z = Temp to:	X = 6 h of RH	X = 8 h of RH	X = 10 h of RH
20 °C	26 °C	Event **A** detected		
13 °C	20 °C		Event **A** Detected	
10 °C	13 °C			Event **A** Detected

successfully validated in Egypt under tomato and potato open field conditions. The developed models resulted in the use a fewer amount of fungicide sprays for effectively management of the selected diseases compared to a routine full-schedule fungicide program with either 7-day or 10-day schedule. More specifically, only 3 to 5 sprays were needed – depending on the crop and disease – while the 7-day and 10-day schedule result in 22 and 15 sprays, respectively. This implies reductions of the used fungicides that varies between 75% and 86.4% depending on the crop and disease. In what follows, we detail our main findings.

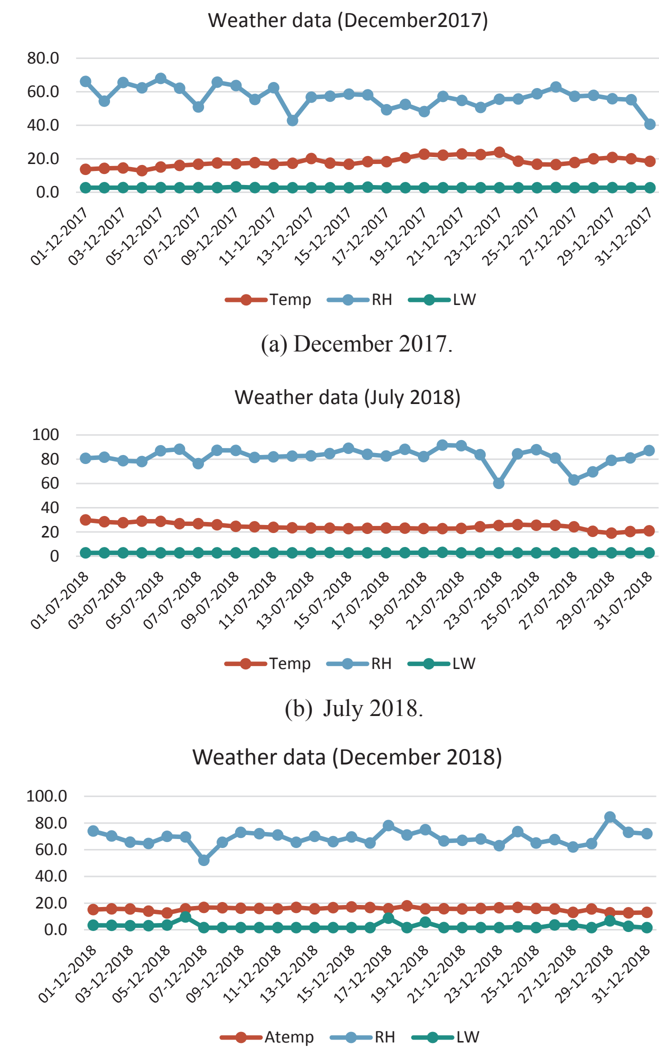
Potato Late blight: The recommended treatment was alerted five times at 5th, 21st of November, and at 1st, 6th, and 23rd of December 2017. Also, it was five times at 9th, 14th of November, and 6th, 17th and 29st of December 2018, respectively. This resulted in having four sprays in 2017 and three sprays in 2018, instead of following a full-schedule fungicide program (routine application) in 2017 and 2018, respectively.

Potato Early blight: Results showed that the disease model has determined the alerts for the first application in 2th, 9th, 21st and 23rd of August 2017. While the recommended treatment was alerted at 4th, 9th and 27th of July; and 6th, 17th, 21st and 26th of August 2018. This resulted in having four sprays in 2017 and 2018, instead of following a full-schedule fungicide program (routine application) in both seasons.

Tomato Late blight: The recommended treatment was given five times at 6th, 11th of Nov., and at 5th, 16th, and 26th of Dec. 2017. While it was four times at 11th, 20th of Nov., and 6th and 24th of December 2018, respectively. This resulted in having three sprays in 2017 and four sprays in 2018.

Tomato Early blight: Results showed that the disease model has accurately determined the correct time for the alerts in 6th, 14th and 23rd of July 2017. The model output daily announcement also successfully detected the disease infection potential in 1st, 16th, and 25th of August 2017. For the second season, the disease infection potential announces were in 6th, 15th, 20th and 21st July 2018. While, the disease model alerts were 5th, 17th, and 23rd August 2018. Therefore, five sprays were applied in the first season and four sprays in the second of evaluation according to the recommendation of the designed model for the warning system.

Tomato powdery mildew: Data illustrated that the model sufficiently detected the powdery mildew daily infection potential and the daily warning announcement for (DIP) was recommended three times



(a) December 2017.

(b) July 2018.

(c) December 2018.

Fig. 14. Sample of the recorded environmental data in during model validation.

Table 6
The basic rules of system analysis to identify event A in the tomato powdery mildew forecasting model.

Y = Temp. From:	Z = Temp. To:	X = 4 h of RH	X = 6 h of RH	X = 8 h of RH
25 °C	29 °C	Event **A** detected		
20 °C	24 °C		Event **A** detected	
15 °C	19 °C			Event **A** detected

at 4th, 11th, 23rd of October, and at 6th, 11th, and 29th of November 2017. While in 2018 it announced five warning massages at 8th, 20th of October and 9th, 16th and 23rd of November. This resulted in having four sprays in both 2017 and 2018 which successfully controlled tomato powdery mildew in 2017 and 2018.

4. Conclusions

In this paper, we have presented the design of an IoT-based monitoring system for epidemic disease control: a key precision agriculture application. The developed system is generic enough to be used with multiple plant diseases where the software architecture can handle different plant disease models. In addition, the used sensors and hardware is carefully designed to be flexible enough to be used with different plants in the monitored fields. An artificial intelligence algorithm has been developed in order to realize an expert system that allows the proposed system to emulate the decision-making ability of a human expert regarding the diseases. It is worth mentioning that the proposed system can be used in future stages to cover the different aspects of precision agriculture such as precision irrigation and automated fertilizer application. The developed system has been used in developing disease models specific for the Egyptian potato and tomato crops based on experiments in order to specify the specific climatic conditions that causes the epidemic disease infection. Such models have been used to verify the expert system software developed for the proposed system.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This work is funded by the Egyptain Science and Technology Development Fund (STDF) under the Demand-Driven Project no. SFTD-DDP-5992. The authors would like to acknowledge the role of the Plant Pathology Research Institute in the Egyptain Agriculture Research Center in facilitating the field trials. The authors would also like to thank Mohammad Abdelilah, Ahmed ElSayed Ali, Nourhan Ehab, and Maha Elgarf for their help.

Appendix A. Supplementary material

Supplementary data to this article can be found online at https://doi.org/10.1016/j.compag.2019.105028.

References

Afifi, M.A., Zayan, S.A.M., 2009. Implementation of EGY-BLIGHTCAST the first computer simulation model for potato late blight in Egypt. Asp. Appl. Biol. 96, 103–110.

Alchourrón, C., Gärdenfors, P., Makinson, D., 1985. On the logic of theory change: partial meet contraction and revision functions. J. Symb. Logic 50 (2), 510–530.

Analog Devices, 2019. ADA4522-1: Single 55 V, EMI Enhanced, Zero-Drift, Ultralow Noise, Rail-to-Rail Output Operational Amplifiers. [Online]. Available at: https://www.analog.com/en/products/ada4522-1.html [Accessed 2019].

Baggio, A., 2005. Wireless Sensor Networks in Precision Agriculture. Stockholm, Sweden. Proceedings of Workshop on Real-World Wireless Sensor Networks (REALWSN'05).

Campbell, P., Bendek, C., Latorre, B., 2007. Risk of powdery mildew (Erysiphe necator) outbreaks on grapevines in relation to cluster development. Cien. Inv. Agr. 34 (1), 1–6.

Durrant-Whyte, H., Henderson, T., 2008. Multisensor Data Fusion. In: *Handbook of Robotics*. s.l.:Springer, pp. 585–610.

Foughali, K., Fathallah, K., Frihida, A., 2018. Using cloud IoT for disease prevention in precision agriculture. Procedia Comput. Sci. 130, 575–582.

Gangwar, D.S., Tyagi, S., Soni, S.K., 2019. A conceptual framework of agroecological resource management system for climate-smart agriculture. Int. J Environ. Sci. Te. 16 (8), 4123–4132.

Hadders, J., 1996. Experience with a late blight DSS (Plant-Plus) in a starch potato area of the Netherlands in 1995 and 1996. The Netherlands. Proceedings of First Workshop of an European Network for Development of an Integrated Control Strategy of potato late blight Lelystad.

HFP Futures Group Making Space for Science - Humanitarian Policy Dialogue, 2011. Unlocking the Potential for Effective Crisis Prevention, Preparedness, Response and Emergency Recovery, s.l.: Humanitarian Futures Programme.

Hossam, M., Kamal, M., Moawad, M., Maher, M., Salah, M., Abady, Y., Hesham, A., Khattab, A., 2018. PLANTAE: An IoT-Based Predictive Platform for Precision Agriculture. Alexandria, Egypt. Proceedings of IEEE Japan-Africa Conference on Electronics, Communications and Computers (JAC-ECC).

Hwang, J., Shin, C., Yoe, H., 2010. Study on an agricultural environment monitoring server system using wireless sensor networks. Sensors 10 (12), 11189–11211.

Ibrahim, H., Mostafa, N., Halawa, H., Elsalamouny, M., Daoud, R., Amer, H., Adel, Y., Shaarawi, A., Khattab, A., ElSayed, H., 2019. A layered IoT architecture for greenhouse monitoring and remote control. SN Appl. Sci. 1 (3), 223.

Intergovernmental Panel on Climate Change (IPCC), 2011. Special Report on Managing the Risks of Extreme Events and Disasters to Advance Climate Change Adaptation.

International Federation of Red Cross and Red Crescent Societies, 2008. Early Warning – Early Action.

Ismail, H., Ehab, N., 2015. Algebraic Semantics for Graded Propositions. Dresden, Germany. Proceedings of 5th Workshop on Dynamics of Knowledge and Belief.

Jaunatre, V., Gaucher, D., 2011. A decision-making aid that evolves. Nouveau Siècle, Lille, France. Proceedings of 4ème Conférence Internationale sur les Méthodes Alternatives en Protection des Cultures Evolution des Cadres Réglementaires Européen et Français Nouveaux Moyens et Stratégies Innovantes.

Jawad, H., Nordin, R., Gharghan, S., Jawad, A., Ismail, M., 2017. Energy-efficient wireless sensor networks for precision agriculture: a review. Sensors 17 (8), 1781.

Khaleghi, B., Khamis, A., Karray, F., Razavi, S., 2013. Multisensor data fusion. Inform. Fusion 14 (1), 28–44.

Khanna, A., Kaur, S., 2019. Evolution of Internet of Things (IoT) and its significant impact in the field of precision agriculture. Comput. Electron. Agric. 157, 218–231.

Khattab, A., Abdelgawad, A., Yelamarthi, K., 2016. Design and Implementation of a Cloud-based IoT Scheme for Precision Agriculture. Cairo, Egypt. Proceedings of IEEE International Conference on Microelectronics (ICM).

Leonard, R., Dowley, L., Rice, B., Ward, S., 2001. Comparison of the NegFry decision support system with routine fungicide application for the control of potato late blight in Ireland. Potato Res. 44 (4), 327–336.

LinkSprite, 2019. SIM900 GPRS/GSM Shield. [Online] Available at: http://linksprite.com/wiki/index.php5?title=SIM900_GPRS/GSM_Shield [Accessed 2019].

Liu, G., Ying, Y., 2003. Application of bluetooth technology in greenhouse environment, monitor and control. J Zhejiang Univ. Agric. Life Sci. 29, 329–334.

Liu, H., Meng, Z., Cui, S., 2007. A Wireless Sensor Network Prototype for Environmental Monitoring in Greenhouses. Shangai, China. Proceedings of International Conference on Wireless Communications, Networking and Mobile Computing (WiCom).

Maia, R.F., Netto, I., Tran, A.L.H., 2017. Precision agriculture using remote monitoring systems in Brazil. San Jose, CA, USA. Proceedings of IEEE Global Humanitarian Technology Conference (GHTC).

Manijeh, K., Amene, D., 2012. A wireless sensor network solution for precision agriculture based on zigbee technology. Wireless Sensor Netw. 25–30.

Microchip Technology, 2019. ATmega2560. [Online] Available at: https://www.microchip.com/wwwproducts/en/ATmega2560 [Accessed 2019].

Morais, R., Fernandes, M.A., Matos, S.G., Serôdio, C., Ferreira, P.J.S.G., Reis, M.J.C.S., 2008. A ZigBee multi-powered wireless acquisition device for remote sensing applications in precision viticulture. Comput. Electron. Agric. 62 (2), 94–106.

Muangprathub, J., Boonnam, N., Kajornkasirat, S., Lekbangpong, N., Wanichsombat, A., Nillaor, P., 2019. IoT and agriculture data analysis for smart farm. Comput. Electron. Agric. 156, 467–474.

Muzafarov, F., Eshmuradov, A., 2019. Wireless sensor network based monitoring system for precision agriculture in Uzbekistan. Telkomnika 17 (3), 1071–1080.

Navarro-Hellín, H., Torres-Sánchez, R., Soto-Valles, F., Albaladejo-Pérez, C., López-Riquelme, J.A., Domingo-Miguel, R., 2015. A wireless sensors architecture for efficient irrigation water management. Agr. Water Manage. 151, 64–74.

Ogalo, L., Bessemoulin, P., Ceron, J.-P., Mason, S.J., Connor, S.J., 2008. Adapting to climate variability & change: the climate outlook forum process. WMO Bull. 57 (2), 93–102.

Rogers, D., Tsirkunov, V., 2011. Costs and Benefits of Early Warning Systems, s.l.: United Nations International Strategy for Disaster Reduction.

Ruiz-Garcia, L., Barreiro, P., Robla, J.I., 2008. Performance of ZigBee-based wireless sensor nodes for real-time monitoring of fruit logistics. J. Food Eng. 87 (3), 405–415.

Shapiro, S., Bona, J., 2010. The GLAIR cognitive architecture. Int. J. Mach. Conscious. 2 (2), 307–332.

Shapiro, S., Group, T. S. I., 2010. SNePS 2.7.1 User's Manual. [Online] Available at: www.cse.buffalo.edu/sneps/Manuals/manual271.pdf.

Shapiro, S., Ismail, H., 2003. Anchoring in a grounded layered architecture with integrated reasoning. Robot. Auton. Syst. 43 (2–3), 97–108.

Shaw, R., Uy, N., Baumwoll, J., 2008. Indigenous Knowledge for Disaster Risk Reduction: Good Practices and Lessons Learned from Experiences in the Asia-Pacific Region, s.l.: United Nations International Strategy for Disaster Reduction.

Shieh, J.-C., Wang, J.-Y., Lin, T.-S., Lin, C.-H., Yang, E.-C., Tsai, Y.-J., Tsai, H.-T., Chiou, M.-T., Lu, F.-M., Jiang, J.-A., 2011. A GSM-based field monitoring system for spodoptera litura (Fabricius). Eng. Agric. Environ. Food 4 (3), 77–82.

Spits, H., Wander, J., Kessel, G., 2003. DSS development focused on variety resistance in The Netherlands 2003. s.l. In: Proceedings of 8th workshop of an European network for development of an integrated control strategy of potato late blight, PPO special report, pp. 215–222.

Trout, C.L., Ristaino, J.B., Madritch, M., Wangsomboondee, T., 1997. Rapid detection of Phytophthora infestans in late blight-infected potato and tomato using PCR. Plant Dis. 81 (9), 1042–1048.

United Nations International Strategy for Disaster Reduction (UNISDR), 2008. Developing Early Warning Systems: A checklist: Third International Conference on Early Warning.

Victoria, L. P., 2008. Combining Indigenous and Scientific Knowledge in the Dagupan City Flood Warning System in Indigenous Knowledge for Disaster Risk Reduction: Good Practices and Lessons Learned from Experiences in the Asia-Pacific Region, s.l.: United Nations International Strategy for Disaster Reduction.

Voltaic Systems, 2019. 9 Watt Solar Panel. [Online] Available at: https://www.voltaicsystems.com/9-watt-panel [Accessed 2019].

Wenting, H., Zhiping, X., Yang, Z., Pei, C., Xiangwei, C., Ooi, S.K., 2014. Real-time remote monitoring system for crop water requirement information. Int. J. Agr. Biol. Eng. 7 (6), 37–46.

Wu, Y., Huang, M., Zhang, X., 2013. Design and implementation of agricultural environment monitoring system based on GIS and SMS/GPRS. Sanya, China. Proceedings of 3rd International Conference on Photonics and Image in Agriculture Engineering (PIAGENG).

Yelamarthi, K., Abdelgawad, A., Khattab, A., 2016. An Architectural Framework for Low-Power IoT Applications. Cairo, Egypt. Proceedings of IEEE International Conference on Microelectronics (ICM).

Zhou, Y., Yang, X., Guo, X., Zhou, M.G., Wang, L., 2007. A Design of Greenhouse Monitoring & Control System Based on ZigBee Wireless Sensor Network. Shangai, China. Proceedings of International Conference on Wireless Communications, Networking and Mobile Computing (WiCom).