10th International Conference on Hydroinformatics
HIC 2012, Hamburg, GERMANY

COUPLING WIRELESS SENSOR NETWORKS AND AUTONOMOUS SOFTWARE FOR INTEGRATED SOIL MOISTURE MONITORING

KAY SMARSLY, KINCHO H. LAW
Department of Civil and Environmental Engineering, Stanford University,
473 Via Ortega, Stanford, CA 94305-4020, USA

Water shortage is a global problem that has severe implications on economic growth and societal well-being, even in the most developed countries. As more than two-thirds of freshwater consumed worldwide are used for irrigation, large quantities of freshwater can be saved by improving the efficiency of irrigation systems. Automatically scheduling irrigation events based on soil moisture measurements is an effective means to reduce freshwater consumption and irrigation costs. This paper presents the design, the implementation, and the validation of an integrated soil moisture monitoring system, which is part of an ongoing research on intelligent irrigation control. The monitoring system consists of a number of wireless sensor nodes that are connected to an Internet-enabled computer system. Autonomous software in the form of self-contained interacting software entities ("mobile software agents") is embedded into the wireless sensor nodes. The autonomous software is designed to precisely trigger irrigation events based on decentralized real-time diagnoses of actual site conditions and external weather information.

INTRODUCTION

The global water use has been growing in the last century more than twice the rate of the population increase [1]. Today, almost one billion people worldwide are living without access to clean freshwater [2]. Main culprits behind this global water problem are excessive water use, inadequate management, population growth, urbanization, and climate change. Not only developing countries, which are facing severe health problems due to limited access to freshwater, but also the world's wealthiest industrial nations are increasingly suffering from water shortages. Increasing water, food and energy prices, and hampered agricultural productivity have major implications on the nations' economies. In the United States, for example, the water prices are growing about 10-15% per year [3].

According to the UN World Water Development Report, 70% of freshwater worldwide is used for irrigation [4]. Conventional irrigation systems usually work on the principle of timer-based irrigation. Timer-based irrigation controllers, incorporated into the irrigation systems, are deployed to trigger irrigation events using mechanical or electromechanical timers [5]. Timer-based systems possess several disadvantages because actual soil and weather conditions are not considered. Consequently, it is likely that the amount of applied water does not match the plant requirements, and either too much or too little water is used for irrigation. Recent studies have unveiled that, on the average, less than 40% of applied water is used by the irrigated plants effectively [4, 6]. Furthermore, it is well known that poorly managed irrigation systems not only contribute to water scarcity, but can also lead to significant soil damage caused by leaching (due to excessive water application) or draining (due to water shortage) [6].

Since early 2000's, "smart" irrigation controllers begun to appear in the market as an alternative to conventional timer-based irrigation controllers [5, 8]. Smart irrigation controllers are systems that automatically trigger irrigation events based on actual site conditions [8]. Soil moisture-based controllers, for example, trigger irrigation events based on the soil moisture content in the root zone of the plants [9]. Ensuring a soil moisture level between the field capacity of the soil and the wilting point of the plants, soil moisture-based controllers typically determine the water requirements by comparing the soil moisture measurements with pre-defined threshold values. Comprehensive reviews of smart irrigation controllers can be found in [10].

Although many smart controllers are capable of timely initiating irrigation events, they lack the ability of automatically adjusting the runtimes, i.e. the quantity of applied water, based on real-time soil moisture measurements. That is, a preset quantity of water is applied for an irrigation event independent of the actual soil conditions [10]. Furthermore, even those smart controllers that calculate start and end times of irrigation events only take measurements from a small number of isolated spots within the observed area (where the sensors are installed) as the basis for decision making, rather than comprehensively considering area-wide site conditions. Consequently, even well-designed and well-managed sprinkler irrigation systems achieve a maximum irrigation application efficiency between 20% and 75% [6].

This paper presents an integrated soil moisture monitoring system that is designed to automatically schedule irrigation events based on real-time soil moisture measurements and actual weather data. For that purpose, wireless sensor nodes are installed in the observed area and connected to an Internet-enabled on-site server. Using different types of sensors, each node monitors a specific region within the area. Most importantly, autonomous software is embedded into the wireless sensor nodes, thereby enabling the nodes (i) to autonomously communicate with each other, (ii) to cooperatively make decisions about irrigation events, and (iii) to dynamically adapt to changing site conditions in real-time. Taking into account the variability of the soil conditions, information from the sensors is automatically assembled to provide an overall view of the soil conditions. Because most calculations are executed directly on the sensor nodes, the data transmission within the wireless monitoring system is significantly reduced. This paper first presents the design and the implementation of the prototype monitoring system. Thereupon, field tests are performed as a proof of concept to illustrate the newly proposed approach and to validate the real-time capabilities of the implemented system. The paper concludes with a discussion of the results and future work.

DESIGN AND IMPLEMENTATION OF A PROTOTYPE SOIL MOISTURE MONITORING SYSTEM

Figure 1 shows schematically the architecture of the prototype monitoring system. The observed area is divided into two regions, labeled “region 1” and “region 2”. Instrumented with soil moisture sensors, wireless sensor nodes are placed in the observed area to continuously collect and analyze the soil moisture of each region. Besides the wireless sensor nodes, an on-site server is included in the system, connected to the wireless sensor network through a base node. This section describes the hardware components of the monitoring system, and delineates the design and implementation of the software embedded into the wireless sensor nodes and installed on the on-site server.

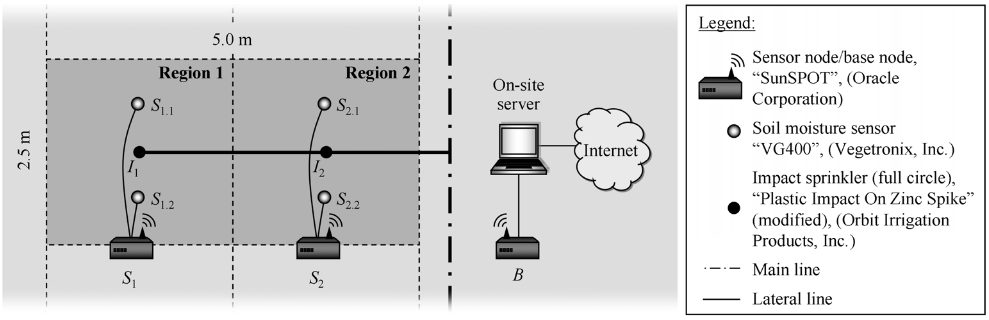

Figure 1. Schematic of the soil moisture monitoring system

Hardware of the soil moisture monitoring system

For the prototype implementation of the monitoring system, Oracle SunSPOT sensor nodes are deployed, one of which serving as the base node [11]. The sensor nodes are built according to the IEEE 802.15.4 standard. For wireless communication, the sensor nodes contain Texas Instruments CC2420 single-chip transceivers that operate on the 2.4 GHz unregulated industrial, scientific and medical (ISM) band. The computational core of a sensor node is a 32-bit ARM920T processor executing at 180 MHz maximum internal clock speed with 16 kB instruction and 16 kB data cache memories. Each node incorporates 4 MB flash memory and 512 kB RAM.

Unlike common embedded applications for wireless sensor networks, which are usually written in low-level native languages (e.g. C/C++ and assembly language), a SunSPOT sensor node comprises a fully capable Java Micro Edition (Java ME) environment, that is widely used, for example, on advanced mobile devices such as smart phones. The most significant feature of the sensor node is that it runs a Squawk Java virtual machine, which executes directly out of the flash memory without an underlying operating system [12]. As a result, memory is saved, which would otherwise be consumed by the operating system. Whereas most Java virtual machines run a single application, the Squawk Java virtual machine can run multiple applications facilitating a concurrent execution of diverse embedded applications implemented for “smart” wireless monitoring.

The sensor nodes, comprising integrated temperature and light sensors, are provided with several analog inputs, general purpose I/O pins as well as high current output pins enabling the connection of external sensors and actuators. For implementing the soil moisture monitoring system, external soil moisture sensors, type Vegetronix VG400 [13], are attached to the sensor nodes through the analog inputs. The VG400 is a low-power soil moisture sensor that senses volumetric water content based on measurements of the dielectric constant of the soil, a technique known to provide highly accurate results. The sensor is insensitive to water salinity and can not corrode over time as, for example, traditional conductivity-based sensors [14].

Embedded software

The autonomous software embedded into the wireless sensor nodes is implemented based on multi-agent technology. Multi-agent systems are composed of self-contained, interacting software entities ("software agents"), which are capable of *flexible autonomous* action executed to achieve their design goals. *Autonomy* of a software agent, according to [15], is defined as acting without the direct intervention of humans, and having control over its internal state. *Flexibility* includes the abilities to (1) react in a timely fashion on events and changes in the environment, (2) pro-actively apply goal-directed behavior by taking the initiative instead of just responding to stimuli, and (3) interact with other agents to perform a task using agent communication languages and appropriate interaction protocols [16].

For carrying out the monitoring tasks in a fully autonomous fashion, software agents are embedded into the wireless sensor nodes, following a mobile multi-agent approach as proposed by Smarsly, *et al.* [17]. Referred to as a "mobile software agent", each software agent is responsible for solving one specific monitoring task directly on the sensor node. As illustrated in the UML class diagram depicted in Figure 2, different types of mobile agents are implemented. On each sensor node, mobile agents are embedded for collecting different sensor data (*soil moisture* and *temperature sensor agent*). Furthermore, a *controller agent* is incorporated on each node to perform on-board data analyses and to communicate both with other sensor nodes and with the on-site server. A controller agent, taking into account current soil and weather conditions, requests sensor data from the other agents and makes real-time diagnoses of the soil moisture conditions. Based on the diagnoses, the agent sends messages to the on-site server to propose starting or stopping of irrigation events. In addition, measurements of soil moisture and temperature are sent to the on-site server by the controller agent in periodic intervals for persistent storage.

The dynamic behavior of the mobile agents is implemented as Event-Condition-Action (ECA) automata composed of states as well as transitions among the states. Transitions are labeled by *ECA* rules, where *E* is an event name, *C* is a condition, and *A* is an atomic action to be taken; an action can be, for example, to measure the soil moisture.

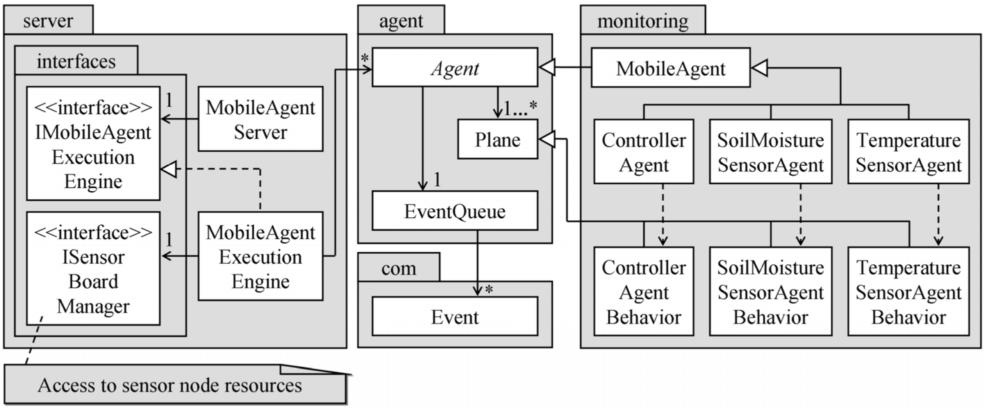

Figure 2. Abridged UML class diagram detailing relevant classes embedded in the wireless sensor nodes

The on-site server is primarily assigned the responsibility for persistently storing the measurement data received from the wireless senor network. The data sets taken from the observed regions are stored in a MySQL database that is remotely available to authorized human individuals. As can be seen from Figure 3, every hour the soil moisture, i.e. the volumetric water content of the soil, and the temperature is taken from the observed regions, sent to the on-site server, and written into the monitoring database. Instead of submitting all collected field measurements to the on-site server, the controller agents are sending averaged values representing the current situation of the observed region. The values are calculated directly on the nodes from multiple measurements resulting in a significant reduction of communicated data. In addition to the field data, weather data is requested from external resources, and automatically integrated into the monitoring database. In the prototype system, the probability of precipitation within the next 24 hours is remotely obtained from an external weather data service. For that purpose, an URL connection to a WWW resource, a dynamic weather website, is created on demand, the website is parsed, and the weather data of interest, here the probability of precipitation, is stored in the monitoring database.

			Time	WaterContentAvg1	WaterContentAvg2	Temperature1	Temperature2	PoP
			2011-10-21 15:00:00	0.095	0.093	22	22.25	0
			2011-10-21 16:00:00	0.094	0.092	22.25	22.25	0
			2011-10-21 17:00:00	0.094	0.092	22.25	22.5	0

Figure 3. Monitoring database (extract)

FIELD VALIDATION TESTS

Validation tests are conducted to study the real-time capabilities and the reliability of the mobile agents. The validation tests also serve as a proof of concept of the newly proposed agent-based monitoring approach.

Validation test setup

In order to test the capabilities of the embedded mobile agents to react appropriately on changing site conditions and to perform cooperative real-time diagnoses of the soil moisture conditions, the prototype system is installed in the field to monitor a 5.0 m × 2.5 m test area. As shown in Figure 1, the area is divided into two monitoring regions. In each region, one wireless sensor node is installed, hosting the mobile agents as described earlier. The wireless sensor nodes (labeled S_1 and S_2) are connected to the on-site server, a laptop computer located next to the test area, through the base node B. Each wireless sensor node is interfaced to two soil moisture sensors and includes one temperature sensor. The soil moisture sensors are placed at a soil depth of 30 cm representing a typical root zone of common plants and crops. Modified impact sprinklers are installed in the middle of each region. Figure 4 shows the instrumentation of sensors in region 2 in the field test area.

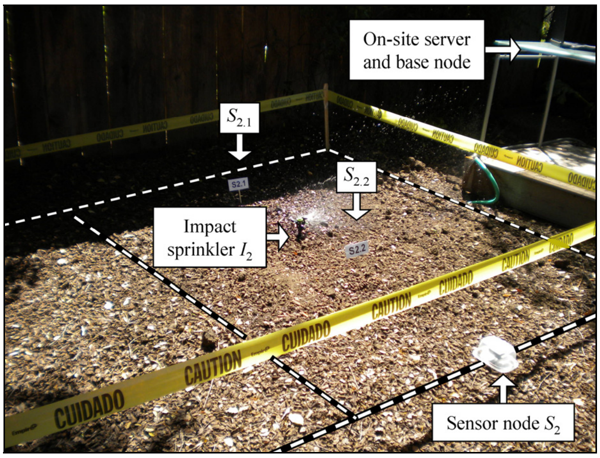

Figure 4. Field tests of the autonomous monitoring system

Autonomous soil moisture monitoring

Prior to the field validation test, the monitoring system is calibrated soil-specifically to ensure accurate soil moisture measurements. Thereupon, the capability of the mobile agents to cooperatively assess and timely react on the varying soil moisture distributions is validated through a simple test procedure. The test procedure is carried out in region 2 of the observed area. First, the range of ideal volumetric water content in the root zone of region 2 is pre-defined as $0.1 \leq \theta_{R2} \leq 0.5$. As soon as the wireless sensor nodes and the mobile agents are launched, the controller agent of wireless sensor node S_2 starts analyzing the actual volumetric water content of region 2. The sensor node requests measurements from the soil moisture sensor agent, which is responsible for sensors $S_{2.1}$ and $S_{2.2}$. As can be seen from Figure 5, the average volumetric water content of region 2 is determined as 0.09.

As a direct reaction to the dry soil, the controller agent sends a message to the on-site server indicating that irrigation of region 2 is required. The irrigation event is executed through the impact sprinkler I_2 using a flow rate of 240 l/h. As shown in Figure 5, the irrigation event is initiated about 150 seconds after the monitoring system is turned on. The controller agent of sensor node S_2, continuously analyzing the volumetric water content, again sends a message to the on-site server after about 1,300 seconds indicating that the average soil moisture in region 2 has reached the desired level, and irrigation is no longer needed.

The test shows that the monitoring system has autonomously captured the spatial and temporal variability of the soil moisture in real-time. The mobile agents have timely reacted to changes in the soil and, as a result of the collaborative real-time diagnoses, messages have been sent to the on-site server to be used for triggering the irrigation event. Last but not least, due to the flexible on-board data processing, the amount of wirelessly transmitted measurement data has significantly been reduced.

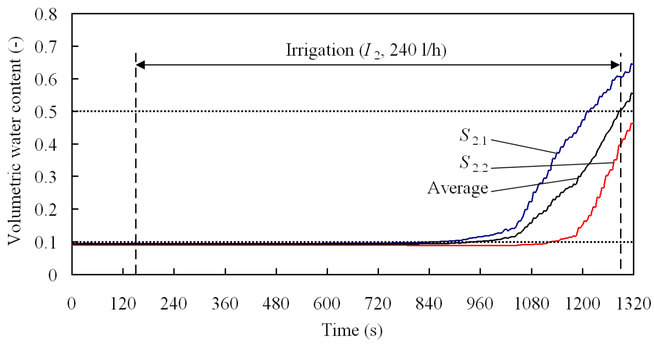

Figure 5. Average volumetric water content of region 2, calculated by the controller agent of sensor node S_2

CONCLUSIONS

The design and the implementation of an autonomous soil moisture monitoring system, which comprises of a wireless sensor network and an Internet-enabled computer system, have been presented in this paper. Field validation tests have corroborated that autonomous software, embedded into the wireless sensor nodes in the form of mobile software agents, can greatly enhance the reliability, the accuracy, and the efficiency of soil moisture monitoring. It has been shown that the mobile agents, autonomously analyzing soil parameters directly on the wireless sensor nodes, are able to reliably assess the spatially and temporally varying soil moisture distribution in real-time; based on interaction and cooperation, the mobile agents respond in a timely manner to changes in the soil enabling a precise scheduling of irrigation events.

Although the prototype system might help to accurately schedule irrigation events and, also, to better understand the soil moisture processes in the environment, the system can further be improved in a number of areas. For example, additional sensors, such as rain sensors, could be integrated into the system. Future work may also include interfacing the autonomous soil moisture monitoring system with an irrigation system comprising automatic valves (actuators) in order to achieve a fully autonomous, integrated irrigation. Further field tests may be conducted to study and to quantify the water reducing potential of the implemented approach.

ACKNOWLEDGMENTS

This research is partially funded by the German Research Foundation (DFG) under grant SM 281/1-1, awarded to Dr. Kay Smarsly, and by the U.S. National Science Foundation (NSF) under grant CMMI-0824977, awarded to Prof. Kincho H. Law.

REFERENCES

[1] UN-Water, "*UN-Water Statistics*", United Nations, (2010).
[2] WHO/UNICEF, "*Progress on Sanitation and Drinking-water – 2010 Update*", World Health Organization, (2010).
[3] Maxwell, S., "Historical Water Price Trends", *AWWA Journal*, Vol. 102, No. 4, (2010), pp 24-28.
[4] UNESCO Publishing, "*The United Nations World Water Development Report 3*", The United Nations Educational, Scientific and Cultural Organization, (2009).
[5] Zazueta, F. S., Smajstrla, A. G. and Clark, G. A., "*Irrigation System Controllers*", University of Florida, Institute of Food and Agricultural Sciences, (2008).
[6] Smajstrla, A. G., Boman, B. J., Clark, G. A., Haman, D. Z., Harrison, D. S., Izuno, F. T., Pitts, D. J. and Zazueta F. S., "*Efficiencies of Florida Agricultural Irrigation Systems*", University of Florida, Institute of Food and Agricultural Sciences, (2008).
[7] California Agricultural Water Stewardship Initiative (CAWSI), "*Irrigation Management*", California Agricultural Water Stewardship Initiative, (2008).
[8] Dukes, M. D., "*Smart Irrigation Controllers: What Makes an Irrigation Controller Smart?*", University of Florida, Institute of Food and Agricultural Sciences, (2009).
[9] Dukes, M. D., Shedd, M. and Cardenas-Lailhacar, B., "*Smart Irrigation Controllers: How Do Soil Moisture Sensor (SMS) Irrigation Controllers Work?*", University of Florida, Institute of Food and Agricultural Sciences, (2009).
[10] U.S. Department of the Interior, "*Weather and Soil Moisture Based Landscape Irrigation Scheduling Devices*", Technical Review Report - 2nd Edition, U.S. Department of the Interior, Bureau of Reclamation (2007).
[11] Sun Labs, "*SunSPOT Theory of Operation – Red Release 5.0*", Sun Microsystems, Inc., (2009).
[12] Simon, D. and Cifuentes, C., "The Squawk Virtual Machine: Java(TM) on the Bare Metal", *ACM SIGPLAN Conference*, San Diego, CA, USA, (2005), pp 150-151.
[13] Vegetronix, Inc., "*Vegetronix VG400 Soil Moisture Sensor Probes*", Vegetronix Inc., (2008).
[14] Zazueta, F. S. and Xin, J., "*Soil Moisture Sensors*", University of Florida, Institute of Food and Agricultural Sciences, (1994).
[15] Jennings, N. R. and Wooldridge, M. J., "*Agent Technology: Foundations, Applications, and Markets*", Springer, (2002).
[16] Wooldridge, M. J. and Jennings, N. R., "Intelligent Agents: Theory and Practice", *The Knowledge Engineering Review*, Vol. 10, No. 2, (1995), pp 115-152.
[17] Smarsly, K., Law, K. H. and König, M., "Resource-Efficient Wireless Monitoring based on Mobile Agent Migration", *Health Monitoring of Structural and Biological Systems 2011*, San Diego, CA, USA, Vol. 7984, 798426-2, (2011).