PRECISION IRRIGATION IN LANDSCAPES BY WIRELESS NETWORK

SLOSSON ENDOWMENT, FINAL REPORT, JULY 2006 – JUNE 2008

PRINCIPAL INVESTIGATOR

Michael J. Delwiche, Biological & Agricultural Engineering, University of California, Davis, One Shields Ave, Davis, CA 95616. (530) 752-7023 (P), (530) 752-2640 (F), mjdelwiche@ucdavis.edu

COOPERATING INDIVIDUALS

Robert W. Coates, Biological & Agricultural Engineering, University of California, Davis, One Shields Ave, Davis, CA 95616. (530) 752-6731 (P), rwcoates@ucdavis.edu
Lorence R. Oki, Plant Sciences, University of California, Davis, One Shields Ave, Davis, CA 95616. (530) 754-4135 (P), lroki@ucdavis.edu
Richard Y. Evans, Plant Sciences, University of California, Davis, One Shields Ave, Davis, CA 95616. (530) 752-6617 (P), ryevans@ucdavis.edu

ABSTRACT

Irrigation control in California landscapes is important since a large portion of the state's available water is used for turfgrass and ornamentals. The diversity of plant species, soil types, and landscape elevations in a single area makes efficient management difficult. The end result is some plants are over-irrigated, wasting water and causing run-off, and other plants are under-irrigated, causing plant distress. Smaller irrigation zones are needed to provide more accurate water delivery. During this 2 year project, we developed wireless valve controllers that self-assemble into a mesh network. Mesh networking means that controllers pass messages to extend the effective communication range without using high power radios. Solar energy was collected with a 200 mW panel to operate each controller node without yearly battery replacement. Nine nodes were tested in a mesh network and each properly responded to commands. Measurements of battery voltage, solar panel voltage, enclosure temperature, and external sensors were transmitted every 10 minutes. Irrigation schedules were stored locally on each node and executed automatically. Schedules for each node were unique, based on the needs of the particular area being irrigated. Internal clock drift was an average 6.3 s per day. Clock offset was removed using daily time stamps. One-hop transmission range using 916 MHz radios varied from 20.9 m with a whip antenna at ground level to 241.1 m with a dipole antenna at 3 m. Node commands were acknowledged after an average of 2.7 s per hop. Charge consumption was approximately 6.76 mA·h per day for the node circuit and 1 mA·h per day for battery self-discharge. The solar panel produced 26.0 to 81.3 mA·h in direct sunlight and 6.5 to 13.7 mA·h in shade. Node operation is expected to be continuous with occasional sunlight exposure. Soil moisture, pressure, temperature, and other environmental sensors will be used for feedback control and detection of problems.

Due to the complexity of the technical design, we have not yet deployed a complete network for landscape testing as of the end date of this grant. With parallel funding from other sources, work is continuing and we expect to test a network for irrigation control and fault

detection before the end of the calendar year. Researchers, farmers, and commercial companies have contacted us regarding our work and we hope that these collaborative opportunities will continue to move this technology toward commercialization.

INTRODUCTION

Conventional irrigation management provides water and nutrients uniformly across an entire field and ignores the reality that demand varies due to differences in soil, topology, and plant water and nutrient status. For site-specific management, large plots are divided into several smaller management units based on variable site characteristics and each is provided individualized water and nutrient input to maximize profits, crop yield, and water-use efficiency, and lessen environmental impacts. The benefits of site specific management have been reported for many years. Matching nitrogen delivery with plant needs has increased fertilizer-use efficiency and net returns in some field crops (Beckie et al., 1997) and reduced nitrate leaching in potato crop simulations (Verhagen, 1997). Variable rate application of granular fertilizer based on individual tree size in citrus reduced overall nitrogen application by 38-40% compared to conventional treatment (Zaman et al., 2005). It seems logical that the benefits of variable rate granular fertilization would be seen for variable rate fertigation as well. Spatially variable management has also been shown to increase profits from corn (Wang et al., 2003; Koch et al., 2004) and improve yield in potatoes (King et al., 2002) and grain sorghum (Yang et al., 2001).

Site-specific irrigation has been most thoroughly tested in center pivot and linear move systems for field crops (Camp et al., 1998; King et al., 1999; King and Kincaid, 2004; King et al., 2005; Kim et al., 2006). Much less development has occurred for fixed irrigation systems which are used in high value permanent crops and commercial horticulture. Site-specific fixed irrigation technology would be applicable in orchards, vineyards, landscapes, nurseries, and greenhouses, each of which has unique management challenges. The water and nutrient demand of trees, plants, and vines are impacted by variations in soil condition, elevation, or microclimate. When applied uniformly, water and fertilizer may leach in light textured soils and pool in heavy soils. Planting on steep slopes, as occurs with some vineyards and orchards, creates difficulty in preventing runoff and maintaining irrigation uniformity due to pressure variations. Commercial nurseries and greenhouses contain many different varieties of ornamental plants in close proximity to one another and must deal with continually changing inventory and strict environmental regulations. A single valve typically controls water flow to many emitters, and if there are plants of differing size or water requirements, some will receive too much water, while others will receive too little. Irrigation control for landscapes in arid parts of the United States is also important since a significant amount of water is used for public turf-grass and ornamentals.

Converting conventional fixed irrigation systems to allow site-specific delivery of water and nutrients would create many small management units, each with a valve that must be independently controlled. Additionally, each should have the capability to read in-field sensors such as temperature and soil moisture, which are commonly used for closed-loop irrigation control. Site-specific control for fixed irrigation systems has been limited. Torre-Neto et al. (2000) used latching solenoid valves to control two laterals per row in a citrus orchard. Each lateral uniformly irrigated half the trees in the row, which were grouped based on size (large and small trees). Miranda (2003) controlled water flow to individual laterals for potted plants based on soil moisture feedback. Coates et al. (2006a, 2006b) and Damas et al. (2001) designed systems to control latching valves and read sensors for irrigation control. In each of these

systems, wiring between valves, sensors, and controllers is expensive to install and is subject to damage by animals and machinery. Miranda et al. (2005) recognized this by developing solar-powered, standalone irrigation controllers with soil moisture sensors. However, the system did not include any communication means for centralized aggregation of sensor data or remote monitoring and reprogramming. Wireless communication has been used to monitor in-field sensors, although many use large batteries and solar panels or still require hard-wired valves for irrigation control.

Recent low-cost, low-power wireless networking technology is well suited to replace wires as the communication medium in many agricultural applications (Gonda and Cugnasca, 2006; Hebel, 2006; Wang et al., 2006). In this paper we describe the development of a solar-powered, wireless network for site-specific application of water, fertilizer, and agricultural chemicals using completely autonomous units with mesh networking capability for both sensing and valve control. Large or small valves can be used to allow management of multiple sprinklers or drip emitters (e.g., laterals), or individual plants or trees (e.g., each microsprinkler). Each valve was programmable with a unique schedule to match differing water and nutrient requirements and could be changed to accommodate replants, disease, growth, or seasonal changes. Data from electrical conductivity, pressure, soil moisture, or flow sensors could allow closed-loop irrigation and fertigation control. In previous work, pressure sensors were used to improve water application accuracy compared to fixed-duration irrigation and provided automatic detection of line breaks and emitter clogging (Coates et al., 2006a).

The specific objectives of this two-year project were to
  (1) design an intelligent valve controller capable of low-power, wireless communication,
  (2) design an energy management system to allow stand-alone operation of each valve controller,
  (3) develop a communication network to link the valve controllers with a central field controller, and
  (4) develop control strategies for applying water and detecting faults.

MATERIALS AND METHODS

SYSTEM DESIGN

Overview

Since this system was intended for application in landscapes, greenhouses, nurseries, and orchards, the wireless network had to be versatile enough to operate in many environments. Mesh networking allows messages to pass from one node to any other node in the network by routing them through intermediate nodes (Figure 1). One advantage of this system is increased network range without using high power radios. This allows greater flexibility in node placement since interference or poor range between two nodes may be rendered moot by an alternate communication path. Another advantage is redundancy; a failed node does not disable the network since multiple routing paths exist. In the system presented here, an operator enters node addresses and irrigation schedules on the central field controller and they are distributed to individual nodes in the network. An optional personal computer can provide a graphical interface, but is not required to operate the system.
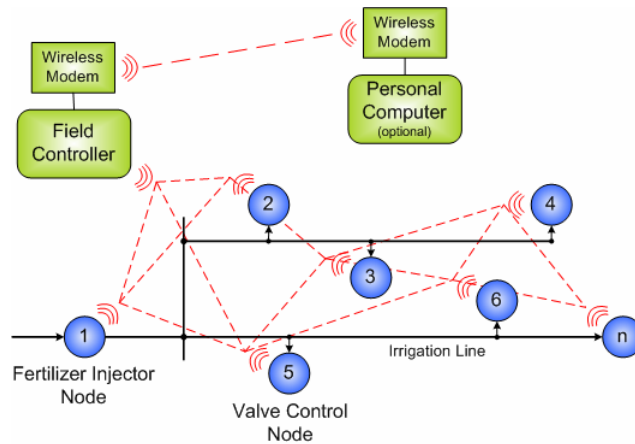
Figure 1. Layout of mesh network for wireless valve control.

Hardware

Bluetooth (IEEE 802.15.1) and ZigBee-based (IEEE 802.15.4) technologies were considered since they have been tested in agricultural environments (Kim et al., 2006; Hebel et al., 2007; Yiming et al., 2007). Bluetooth was deemed not suitable for this development due to its higher energy consumption, shorter range, and lack of support for mesh network routing (Baker, 2005; Hebel, 2006; Wang et al., 2006; Lee et al., 2007). A custom-built system which would require only a microcontroller and radio transceiver was not selected due to the complexity of implementing robust mesh networking software. Instead, commercially available low-power, mesh networking technologies were tested.

Our first-generation prototype for a wireless microsprinkler was designed using ZigBee demonstration boards (PICDEM Z, Microchip Technology, Chandler, Ariz.). The mesh network communication protocol was handled by the company's implementation of the ZigBee wireless networking standard (*http://www.zigbee.org*). At that time, we found that the ZigBee implementation did not support battery-powered routers that can sleep between radio communications. While mesh networking is a key feature in ZigBee, routers are generally required to have main-line power. There is a provision allowing for 'beaconed' networks (time-synchronized networks that allow routers to sleep), but no ZigBee vendors could be found that had implemented it in their software. It was decided to use different technology instead of attempting to implement time-synchronization in this system.

Our second generation prototype used low-power wireless modules (Tmote Sky, Moteiv, San Francisco, Cal.) designed specifically for battery-powered mesh networking. A low-power wireless module like this is commonly called a "mote," which is defined as a small particle or speck, because it is the result of research aimed at sensors only a cubic millimeter in size (Warneke et al., 2001). The modules were programmed with TinyOS (*http://www.tinyos.net*), an open-source operating system written for wireless sensors. TinyOS includes its own communication protocol, but ZigBee-compliant modules running TinyOS are being developed (Suh and Horton, 2004). ZigBee compliance would provide the benefits of industry standardization, such as vendor product interoperability, security, and marketability. The second generation prototype was tested using two communication protocols: a broadcast messaging component (Drip) was used when sending downstream messages from the field controller to the

valve controllers, and a mesh routing routine (Multihop) was used for sending upstream messages from the valve controllers to the field controller. Testing of the mesh network showed that sending messages from the field controller to the valve controllers was not as reliable or efficient as expected.

Another TinyOS-based wireless module, the MICA2, (MPR400CB, Crossbow Technology, San Jose, Cal.) was adopted for our third generation valve controller design, to be discussed here (Figure 2). The reasons for moving to these modules were that the mesh networking software, XMesh, still uses TinyOS but includes improved downstream messaging, and the company is interested in developing products for agricultural monitoring and control, thus providing a good opportunity for collaboration and increased likelihood of future commercialization. The wireless modules used here operated at 916 MHz.

A prototype circuit board was provided by Crossbow Technology for development of the valve controller (remote node). It included a 51-pin connector to interface with the wireless module, a 3.0 V voltage regulator, thermistor, signal multiplexer, and input/output (I/O) screw terminals. Figure 3 shows a simplified block diagram of the circuit components and connection to valve and sensors. Valve actuation and sensor excitation were controlled by microcontroller outputs on the wireless module. The input channel multiplexer allowed a single analog-to-digital converter (ADC) input on the microcontroller to measure each of eight signals in succession, including board-level temperature, solar panel voltage, and 6 external sensors. A second ADC input measured battery voltage.

A 7.2 V, 170 mAh nickel-cadmium battery constructed from two 3.6 V batteries in series (BattR5, Solarbotics, Alberta, Canada) and a 200 mW (13.4 V, 15 mA) solar panel constructed from two 6.7 V panels in series (SCC3733, Solarbotics, Alberta, Canada) were selected to provide continuous operation without yearly battery replacement. The solar panel was selected to provide higher voltages than the expected 9 V maximum battery voltage during charging, and current that would supply enough charge to replenish all energy used by the node. A nickel cadmium (NiCd) battery was chosen because it is more resilient to overcharging, is generally less expensive than nickel metal hydride (NiMH) and lithium ion (Li-ion) battery chemistries, and has a lower internal resistance for providing pulses of high current with less voltage drop (Linden and Reddy, 2001). The disadvantages of NiCd are self discharge of about 15-20% per month, larger size than NiMH and Li-ion, and disposal restrictions. With shallow discharge from the low-power circuitry used here, the NiCd battery is expected last for several thousand daily cycles (Linden and Reddy, 2001).

Many wireless sensors are designed to operate on only 3 V, but a higher level was used here in order to provide adequate voltage for operating a 1-inch or 1/8-inch latching solenoid valve (Netafim, Tel Aviv, Israel). The valves were rated for 12 VDC or more, but operated effectively at 7.2 V. Valve control voltage could easily be boosted by using a slightly larger battery or charge pump with storage capacitors. Bidirectional current to the valves was controlled using an H-bridge switching circuit composed of two N-channel metal-oxide-semiconductor field-effect transistors (MOSFETs) and two P-channel MOSFETs. Two more N-channel MOSFETs inverted the microcontroller signals to drive the P-channel MOSFETs. Four diodes were connected to the valve to provide suppression of inductive voltage spikes produced when turning off the MOSFETs. Valves were opened or closed with an 80 ms pulse from the battery.

The wireless modules had an MMCX jack used with a 1/4-wave whip (provided with the module) or 1/2-wave dipole antenna (S467FL-5-RMM-915S, Nearson, Springfield, Va.). The

circuit components were housed in a clamshell-style polycarbonate enclosure (1030 Micro Case, Pelican Products, Torrance, Cal.) to provide dirt and moisture protection during outdoor testing (Figure 2). Holes were drilled in the box for the antenna, valve, and sensor wires. The antenna was mounted directly to the box and the internal wire hole was sealed with silicone. Cable ports were used to provide a seal for valve and sensor wires.

A base node consisting of a wireless module and RS-232 gateway (MIB510CA, Crossbow Technology) was connected by serial cable to an embedded controller (TD40, Tern, Davis, Cal.), which served as the field controller for the network of remote nodes. The field controller contained a keypad to allow entry of schedules and manual operation of the remote valves, and a liquid crystal display (LCD) for viewing status information (Figure 4). Several keys were labeled with functions: Time, Schedule, Manual Control, Node Status, Enter, and Delete. The remaining keys were numbered zero to nine for input of addresses, time, commands, etc. Other details of the field controller were described by Coates et al. (2006b).

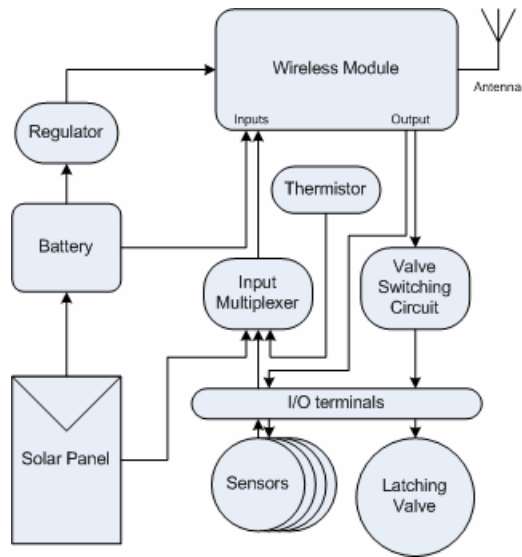**Figure 2. Third generation valve controller with 1-inch latching valve.**

Figure 3. Block diagram of valve controller (node) primary components.

Figure 4. Field controller with embedded controller and wireless module (base node).

Software

The mesh networking protocol (XMesh) was handled by software included with the wireless modules. Formation and operation of a mesh network was as follows. When powered, the nodes automatically began forming a network by transmitting 'route update' messages. Route update messages were broadcast by each node so that neighboring nodes could determine the wireless link qualities between each other. This information was used to determine the best path for message routing from a remote node (valve controller) to the base node (field controller).

Every remote node created a routing table that included an entry for each neighboring node. Associated with each neighbor was a routing cost metric, based on the shortest path-to-base and link quality indicators from the route update messages. A remote node transmitted data messages to the neighbor with the lowest cost (called its parent). If a transmission failed, the message was re-routed to the neighbor with the next lowest cost. Route update messages were transmitted every 5 minutes to ensure routing tables in each node were updated as network conditions changed, reducing the likelihood that impromptu re-routing would be necessary. Once these routing paths were established, messages could be sent upstream from remote to base or downstream from base to remote. Upstream messages were passed (hopped) from one node to the next along the best route until the base node was reached. Downstream messages simply hopped between nodes along the reverse route of the upstream messages until the destination node was reached. The base node connected to the embedded controller was programmed to run the manufacturer's software (XMesh-Base).

The remote nodes were programmed with code written in nesC, an extension of the C language used for programming with TinyOS. The primary features used for irrigation control and sensing were valve actuation routines, a software real-time clock, schedule storage and execution, and individual sensor measurement routines (Table 1). Simple valve commands (i.e., open, close, toggle valve) were sent using the XCommand messaging component provided by the manufacturer. A CustomCommand message type was created for all other commands. Valve operations were triggered by an XCommand message, CustomCommand message, or entry in the stored schedule. All other commands were triggered by a CustomCommand message or schedule entry. OpenValve, CloseValve, and ToggleValve simply actuated the valve accordingly. IrrigationCycleTime opened the valve, waited for the specified duration to elapse, and then closed the valve. The SetClock command was used to send time stamps to synchronize the remote node clock with the embedded controller. Time stamps were automatically transmitted to nodes when joining the mesh network and once each day. GetClock caused the remote node to send the current time of its software clock. ReadSoilMoisture, ReadPressure, ReadOnboard, and ReadAllSensors caused the appropriate sensors or onboard voltages (i.e., solar panel, battery, thermistor) to be measured and returned in a message to the field controller. These four commands were not used in tests presented here. Instead, all sensor inputs and internal voltages were transmitted by the remote node every 10 minutes. ScheduleInterval caused a command to be executed at a recurring interval. For example, a sensor reading could be scheduled to occur every 30 minutes or an irrigation cycle every 4 days. ScheduleDayOfWeek executed a command at a specific time on one or more days of the week. For example, an irrigation cycle could be scheduled to run every Tuesday and Saturday at 7:00 a.m. ScheduleOneTime scheduled a single execution of a command to occur at a later time and date. Since the objective is for the remote nodes to run continuously for years without loss of power, schedules were stored in RAM. Schedules could be stored in the non-volatile EEPROM, but at the expense of slower access time and higher energy consumption. Scheduled commands were deleted individually or all at once using the ScheduleDeleteOne or ScheduleDeleteAll commands.

Each remote node in the network was programmed with a unique address between 1 and 9999 (XMesh allows addresses between 1 and 65534, excluding 126). Upon power-up, the node initialized variables and started timers that control how often internal events were triggered (Figure 5). The node then operated in an event-driven fashion - functions were called when triggered by an interrupt event (e.g., received a radio message, a timer expired). When a

SetClock command was received, the time stamp included with the message was assigned to the clock variables of the node. Each second, an internal timer caused the clock variables to be incremented by one second. When a valve or sensor message was received by the node, it was parsed and the command was pushed into a pending-command queue for execution. Schedule messages caused the command and related data parameters to be stored in memory. Immediately after processing any radio message, an acknowledgement was transmitted back to the base node. This only acknowledged that the command was received, but did not confirm its proper execution (e.g., that the valve actually opened properly). Once per minute, the internal schedule was checked for entries requiring execution (i.e., the scheduled time matched the clock time) and the associated commands were added to the pending-commands queue. Every 10 minutes, a sensor timer triggered the measurement of sensor inputs and internal voltages. These values were transmitted to the field controller. Closed-loop irrigation control based on soil moisture measurements and fault detection based on pressure measurements were tested in previous work (Coates et al., 2006a) and will be added in future software revisions. Closed-loop fertigation control using conductivity sensors and flow meters is also being investigated.

Table 1. Remote node command descriptions.

| Group	Command	Trigger(s)	Function
Valve Operation	OpenValve	XCommand Msg, Schedule	Latch valve open
	CloseValve	XCommand Msg, Schedule	Latch valve closed
	ToggleValve	XCommand Msg, Schedule	Latch valve to opposite state
	IrrigationCycleTime	CustomCommand Msg, Schedule	Open valve, Wait for t minutes, Close valve
Clock	SetClock	CustomCommand Msg	Set remote node clock to match field controller
	GetClock	CustomCommand Msg	Get remote node clock time, transmit to field controller
Command Scheduling	ScheduleOneTime	CustomCommand Msg	Schedule a command for a specific date/time
	ScheduleInterval	CustomCommand Msg	Schedule a command for a specific time and every t minutes thereafter
	ScheduleDaysOfWeek	CustomCommand Msg	Schedule a command for specific weekdays/times
	ScheduleDeleteOne	CustomCommand Msg	Delete one previously scheduled command
	ScheduleDeleteAll	CustomCommand Msg	Delete all previously scheduled commands
Sensor Measurement	ReadAllSensors	CustomCommand Msg, Schedule	Measure all sensors, transmit values to field controller
	ReadSoilMoisture	CustomCommand Msg, Schedule	Measure soil moisture sensor, transmit value to field controller
	ReadPressure	CustomCommand Msg, Schedule	Measure pressure sensor, transmit value to field controller
	ReadOnboard	CustomCommand Msg, Schedule	Measure temperature & battery/solar voltages, transmit values to field controller
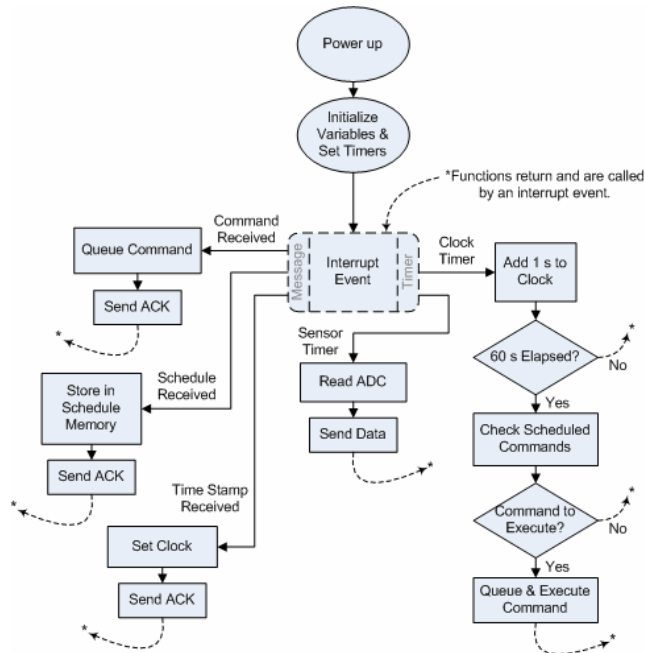
Figure 5. Remote node software flow diagram.

Software was written in C++ for operation of the embedded controller. Commands were entered on the field controller (embedded controller keypad) and sent by RS-232 serial port to the base node and then by radio to the remote nodes. While idle, the field controller displayed the number of active nodes, indicated when a new node had joined the network, or displayed an error message when a node had not responded for more than 1 hour. Four function keys were used to enter specific modes of operation (Table 2). When a function key was pressed, the controller simply prompted the user to select from a menu of commands, enter a remote node address, and enter any specific data associated with the selected command. Commands properly received by the remote node triggered an application-level acknowledgement which was sent back to the embedded controller. Success or failure to receive this acknowledgment was displayed on the embedded controller display.

Table 2. Field controller function key descriptions.

Function key	Description
Time	Change field controller clock of view/update remote node clock
Manual Control	Send command to remote node for immediate execution
Schedule	Schedule one-time or recurrent commands
Status	View remote node data or time of last message

Wireless Range and Function

Maximum one-hop radio range was tested using the base node and one remote node. A simple program was written in which the remote node transmitted seven messages to the base node and summed the number of acknowledgements received in return. The sum was displayed in binary on three LEDs of the remote node and audibly signaled on a piezoelectric buzzer by varying duty cycle. The acknowledgement status at the remote node was monitored while moving further or closer to the base node. At each location, the remote node was rotated around the axis of its antenna. Maximum range was defined as the distance between the two nodes when six or more acknowledgements were consistently received. In most cases, all seven messages were consistently acknowledged at the recorded maximum range and moving further away resulted in frequent fluctuations below six acknowledgements. The wireless nodes were tested with 1/4-wave whip antennas and 1/2-wave dipole antennas. Tests were conducted under visual line-of-sight conditions (open field) and obstructed conditions (young peach orchard with 4 m high canopy) with the nodes on the ground or elevated 1, 2, or 3 m on a wooden stake. For each antenna/field combination, three measurements were taken (moving away from the base in different directions for each) and the mean range was calculated.

The general functionality of network messaging was tested by sending valve, time, and schedule commands to the remote nodes in a mesh network. Eight remote nodes were placed close to the base or in distant locations that forced them to create a multi-hop mesh network. Confirmation of mesh connectivity was obtained by monitoring radio traffic near the base node. The GetClock command was transmitted to remote nodes at one (nearest base), two, and three hops away. Acknowledgement and upstream transmission of the clock time after each command indicated correct functionality of the nodes and network. OpenValve and CloseValve commands were also transmitted to remote nodes with valves. Message acknowledgement and correct valve actuation were noted. The delay between issuance of the command and receipt of acknowledgment at the base was used to estimate the per-hop time required for completion of messages. Automatic re-routing of messages was tested by turning off one node along the preferred routing path, thus forcing a new path to be selected. Re-routing was successful if messages still reached their destination and resulted in an acknowledgement.

Proper schedule execution required the remote nodes to maintain the correct date and time during operation. However, the internal clock was subject to inaccuracy because of crystal frequency drift. For a ±30 ppm rated crystal, this would amount to about ±2.6 seconds per day. A simple test was done to measure the amount of daily clock drift. To start, the clock of the embedded controller was set to the nearest second using a reference time provided by a U.S. government website (NIST and USNO, 2008). While not an official NIST standard when accessed through the website, it claims to be accurate within 0.1 s. The clocks of two remote nodes were set by radio transmission of the current time stored on the embedded controller. Over eight days, the clocks of the embedded controller and the two remote nodes were queried and compared to the reference time. The average clock drift per day was calculated. Any offset introduced by radio delay when setting the remote clock was theoretically canceled out by the same delay when the clock was queried. For example, say the embedded controller transmitted the current time, t_1, to a node. If there was a 2 s radio delay, the remote node had local time t_1-2 relative to the embedded controller. At time t_2, the embedded controller transmitted a clock query to the remote node. After the 2 second radio delay, the embedded controller time was

t_2+2, and the remote node was just receiving the message and returned its local time of t_2 (i.e., the same time at which it was queried).

Energy Management

Current consumption of the remote nodes was measured by reading the voltage drop across a 10 Ω resistor placed in series with the battery. The resistor had a relatively large value, but allowed greater resolution when monitoring small currents. The greater voltage drop it produced did not induce low-voltage problems with the circuit since the battery voltage (7.2 V) was far above the circuit operational voltage of 3.0 V. Voltage drop was viewed on an oscilloscope during wake/sleep power cycling, radio operations, sensor measurement, and valve operation. The resulting waveforms were used to estimate current and duration for each operation. A Riemann sum with 20 μs interval was used to calculate charge consumed for power cycling. Each wake-up was brief (about 7 ms) and capacitors supplied most of the momentary current, not the battery, so measured voltage drop did not reach the expected peak current for a radio operation. After cessation of the wake cycle, the battery recharged the capacitors over 23 ms. For each operation, abrupt current changes resulted in nearly square waveforms, so charged consumed per day was estimated by

$$q = \frac{itn}{10^6} \tag{1}$$

where q was the charge consumed per day (C), i was the current (mA), t was the duration of a single operation (ms), and n was the number of times the operation occurred each day. NiCd batteries self-discharge at 15-20% per month (Linden and Reddy, 2001), which for the 170 mA·h battery used here, was about 29.75 mA·h per month (1 mA·h per day or an average current of 42 μA). The total estimated node charge consumption plus self discharge was compared to solar panel charge production.

Solar panel performance was tested in full sunlight and full shade conditions. A data logger recorded open-circuit voltage from one panel and output current as the voltage drop across 10 Ω load resistors for two panels every 3 minutes over 17 days. The open-circuit panel and one load-connected panel were mounted horizontally in direct sunlight. The second load-connected panel was mounted horizontally in the shadow of an opaque board to produce complete shading from direct sunlight. Indirect sunlight could still illuminate the panel, but all sources were relatively non-reflective surfaces such as buildings. The total charge produced each day was estimated using a Riemann sum with a 3 minute interval.

Solar charging of the Ni-Cd battery was checked using two valve controller nodes, one with a solar panel and one without. The node enclosures, which had a clear lid, were covered with paper to reduce direct solar heating of the internal components. They were placed outside for nine sunny days and set to transmit data messages every 15 seconds (to increase energy use). The messages contained measurements for microcontroller voltage, battery voltage, solar panel voltage, thermistor resistance, and sensor inputs. One message was logged at the field controller every two minutes. Battery and thermistor measurements were converted from ADC integers to voltage and temperature. Battery voltage and temperature inside the enclosure were plotted over time. Theoretical battery life without recharging was calculated by dividing the battery capacity by the node charge consumption. This assumed the battery fully discharged its rated capacity. A more realistic battery life estimate was made by operating a node without solar panel until its voltage fell below the nominal 7.2 V level.

RESULTS AND DISCUSSION

WIRELESS RANGE AND FUNCTION

The radio range results (Table 3) show that mean range varied greatly depending on the node configuration and the test environment. Orchard range was difficult to measure due to erratic changes in acknowledgement status near the maximum range. In some cases, the remote node could be moved a few inches between two locations, causing the number of acknowledgements to change from zero to seven. This illustrates the value of a mesh network, which provides multiple paths of communication. To achieve a one-hop range of 100 m would, in the conditions tested here, require a dipole antenna mounted slightly higher than 1 m or a whip antenna mounted slightly higher than 2 m. In general, elevating a node with whip antenna by 1 m improved range as much as adding a dipole antenna. Ground level units would require about 20 to 30 m spacing to ensure adequate wireless connectivity. Extrapolation of these results for range estimation in other fields or orchards would be difficult since the conditions would likely be different in each location.

Table 3. Radio range under various conditions.

View	Antenna	Elevation (m)	Mean Range (m)
VLOS[1]	Whip	0	20.9
		1	67.6
		2	97.8
		3	205.2
	Dipole	0	32.7
		1	92.8
		2	192.6
		3	241.1
Orchard	Whip	0	21.7
		1	46.9
		2	94.0
		3	119.4
	Dipole	0	30.0
		1	83.2
		2	128.4
		3	145.9

[1]Visual line-of-sight

Careful node placement during network deployment, even if not ideal, can help ensure good connectivity. Using a start-up sequence similar to our seven-acknowledgement test would allow radio communication quality to be assessed during installation. Also, IEEE 802.15.4 compliant wireless modules operating at 2.4 GHz have become more popular among manufacturers. These modules have improved range due to increased sensitivity, and could be adopted as replacements for the current 916 MHz models.

In the mesh network test, each node acknowledged commands, returned correct clock values, and properly opened or closed a valve. This indicated that the mesh network was operating correctly, although there were a few instances where nodes did not respond on the first

attempt. This always occurred after moving the nodes to a new location or early in the re-routing test. In all cases, waiting several minutes allowed the network to stabilize and operate correctly. The average time between command and acknowledgment at the base was 2.2 s for a one-hop message, 5.6 s for a two-hop message, and 9.3 s for a three-hop message, giving a mean of 2.7 s per hop. In the re-routing test, one node in the path to the furthest node was turned off. Initially, the downstream path was broken and commands received no acknowledgement from the remote node. Since downstream messages must follow the path formed by upstream messages, a new path was not established until upstream data and health messages initiated the re-routing process (i.e., selection of a new parent to replace the removed node). After several minutes of no response, subsequent downstream messages were successfully routed along the new path. An acknowledgement and correct time data were received from the destination node.

Figure 6 shows the clock drift for the embedded controller and two nodes over 8 days. The clock of each remote node lagged the reference time after only half a day. A linear regression of the embedded controller data gave an average lag of 0.4 s per day. A linear regression of the combined data of nodes 1 and 2 gave a lag of 6.3 s per day of operation. This was longer than expected, possibly due to greater than rated crystal skew. To ensure the embedded controller and remote nodes maintained synchronized clocks, the remote node clocks were updated with a new time stamp each day. Based on the per-node message delay in one direction (half the measured command-to-acknowledgement delay), the clock of a node 5 hops from the base would lag 6.8 s upon initialization. For synchronized power-cycling of the network, the remote nodes calculate clock skew and offset to maintain an accurate internal clock (not based on real-world time). While this method may be implemented for the real-time clock, an accuracy of several seconds is adequate for irrigation control.
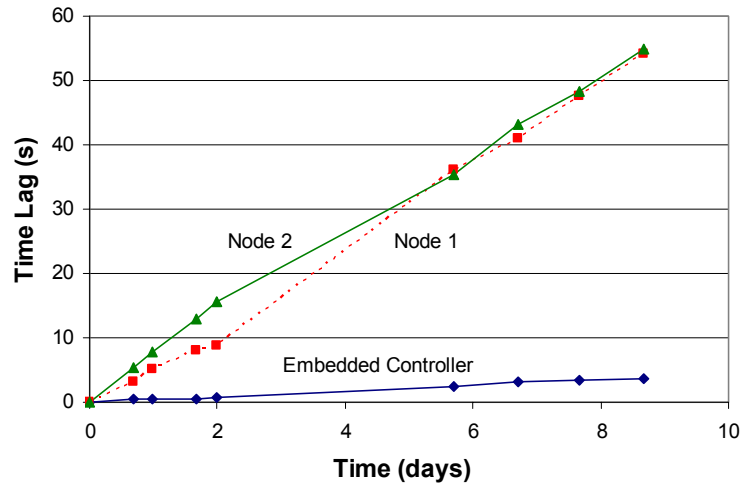

Figure 6. Time lag of remote node clocks and embedded controller clock compared to reference time.

ENERGY MANAGEMENT

Current and consumed charge for each valve controller operation are shown in Table 4. To extend battery life, nodes were in sleep-mode most of the time and only used the radio when

data transfer was required. This power-cycling feature was included with the wireless module software. The nodes spent most time in sleep-mode and synchronously woke every 125 ms to listen for radio activity. If no activity was detected, the node returned to sleep. The average sleep current of 66 µA was higher than the wireless module's rated current of 15 µA due to several resistor bridges on the circuit boards used to measure solar panel and battery voltages, and to set regulator output voltage. Other sensors, such at the thermistor and external sensor inputs were connected to digital outputs and were disabled when not in use. Additional circuit modifications could reduce sleep current further. The current consumption for sensor measurements was based on the specifications of an integrated pressure sensor. The total charge consumption of a node was estimated to be 24.35 C per day, which is equivalent to 6.76 mA·h per day and a mean current of 0.28 mA. Several radio communication parameters could be modified to decrease the frequency of radio use and save additional energy. Node charge consumption and battery self-discharge has to be balanced by solar panel charge production in order to ensure continuous operation of the valve controller.

Table 4. Current, duration, frequency, and calculated charge consumption during remote node operations.

Operation	Peak Current	Duration (ms)	Frequency	Charge Consumed/Day
Sleep	0.066	If not awake	If not awake	4.78
Power cycling (RX)	4.1[1]	7[1]	Every 125 ms	16.31
Route Msg (TX)	15	150	Every 5 min	0.648
Route Msg Neighbors	14	100	Every 1 min[2]	2.016
Data Msg (TX)	15	40	Every 10 min	0.086
Command Msg (RX)	14	120	Every 6 hours	0.007
Command Ack Msg	15	40	Every 6 hours	0.002
Health Msg (TX)	15	50	Every 7 min	0.154
Health Ack Msg (RX)	14	80	Every 7 min	0.231
Valve Actuation	615	80	Twice per day	0.098
Sensor Measurement	7	20	Every 10 min	0.020
TOTAL				24.35

[1]Because of short duration, capacitors supplied most momentary current, clipping the measured voltage drop at 41 mV; battery then recharged capacitors over 23 ms.
[2]Route messages were assumed to come from 5 neighbors at 5 min intervals resulting in an average of one per minute.

Figure 7 shows the open-circuit voltage in full sunlight and calculated output currents from solar panels in sunlight and shade for four typical days. Peak open-circuit voltage was 12.7 V and peak current was about 15 mA in direct sunlight and 1.5 mA in shade. Integration of current over a single day yielded a minimum and maximum charge production of 93.6 and 292.7 C (26.0, 81.3 mA·h) in direct sunlight and 23.4 and 49.3 C (6.5, 13.7 mA·h) in shade. Full sunlight on a daily basis would overcharge the battery, whereas full shade on a daily basis might not provide adequate energy to recharge it. Energy management will be critical to ensure continuous node operation without battery degradation from overcharging. A refined design could include a transistor switch to disconnect the solar panel when battery overcharging occurs. Minimizing radio use during periods of poor energy production could be enabled through software. Jiang et al. (2005) described an energy management scheme using supercapacitors and a Li-ion battery that could also be adopted.

Figure 8 shows the battery voltage and enclosure temperature for the solar recharged and non-recharged nodes. It is evident by the voltage peaks that the solar panel charged the battery each day. However, the daily voltage low-point started to decrease after day 5. We think this was due to overcharging of the battery and excessive heating of the enclosure. A daytime enclosure temperature over 40 °C resulted in a depressed battery voltage the following morning, likely due to a combination of charging at high temperatures and depressed operating voltage in the hot afternoon and cold morning (Linden and Reddy, 2001). Solar radiation shielding of the enclosures will be necessary to protect the circuit and battery. Continued operation of this node following the test indicated that the battery did not suffer from a permanent depression in voltage. However, the long term capacity of the battery may be reduced. The battery voltage of the non-recharged node slowly decreased through the duration of the test with fluctuations due to changes in battery voltage and resistance during daily temperature variation.

Theoretically, the 170 mA·h NiCd battery used here should be able to supply a 7.76 mA·h per day load (node and self-discharge) for 22 days. In testing, this duration was not achieved. A node without solar panel operated for just over 13 days before its battery voltage fell below 7.2 V, and finally to 3.5 V after a total of 17 days. One reason for the less than theoretical operating duration is a difference in the load used here compared to that used for battery capacity determination.
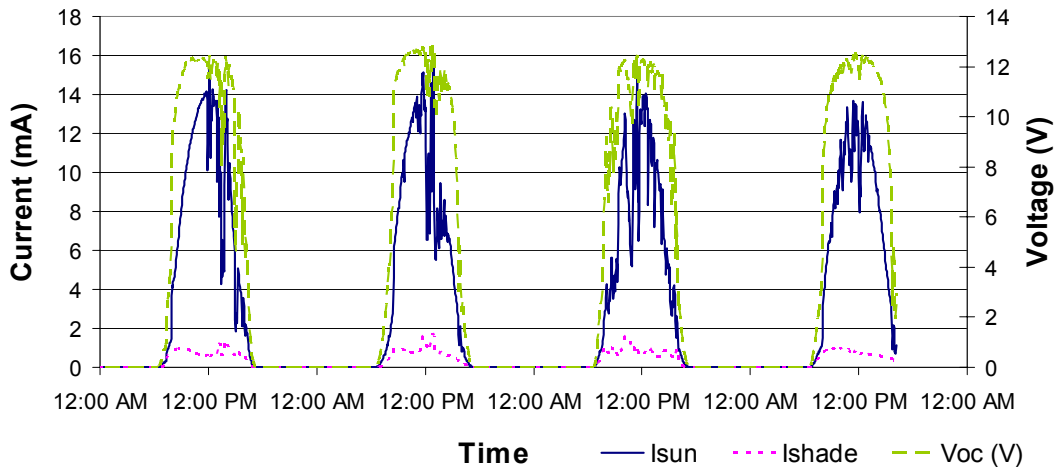

Figure 7. Solar panel open-circuit voltage and current through 10 Ω load over 4 of 17 test days.
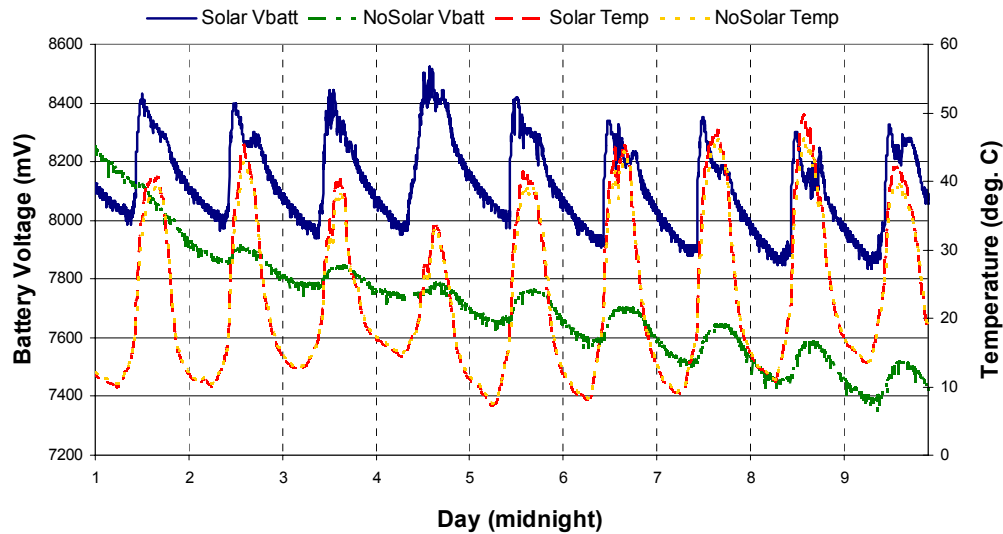
Figure 8. Battery voltage and temperature inside valve controller enclosures over 9 days.

CONCLUSIONS

A wireless, solar-powered valve controller network was designed and tested. A field controller provided a simple interface for monitoring and control. Mesh network communication using 900 MHz radio modules was successful in allowing transmission and acknowledgement of commands for valve operation, scheduling, sensor measurement, and a remote node clock. Messages were sent to nodes up to three hops away with a mean command-to-acknowledgment time of 2.7 s per hop. Clock drift of 6.3 s per day was removed with daily time updates. Mean one-hop radio range was between 20.9 m and 241.1 m, depending on antenna type, elevation above ground, and surrounding environment. We recommend using a dipole antenna and node mounted greater than 1 m high. A multi-hop mesh network allows greater network coverage even when one-hop range is short. Energy management evaluation showed that a 200 mW solar panel produced at least 6.5 mA·h per day in shade and 26.0 mA·h per day in direct sunlight. With a total charge consumption of 7.76 mA·h per day, there should be adequate solar energy produced to continuously power the node if direct sunlight is available part of the day. Solar radiation shielding is needed to reduce enclosure overheating. Energy-use monitoring could further reduce node charge consumption and a solar panel disconnect could prevent battery overcharge.

A network of wireless valve controllers will be deployed for field testing of irrigation control and fault detection. Additional opportunities for academic and commercial collaboration will be explored with the goal of further refining the technology, reducing cost, and commercializing a reliable product for landscapers, farmers, and businesses.

REFERENCES

Baker, N. 2005. ZigBee and Bluetooth strengths and weaknesses for industrial applications. *Computing & Control Engineering* 16(2): 20-25.

Beckie, H.J., A.P. Moulin, and D.J. Pennock. 1997. Strategies for variable rate nitrogen fertilization in hummocky terrain. *Canadian Journal of Soil Science* 77: 589-595.

Camp, C.R., E.J. Sadler, D.E. Evans, L.J. Usrey, and M. Omary. 1998. Modified center pivot system for precision management of water and nutrients. *Applied Engineering in Agriculture* 14(1): 23-31.

Coates, R.W., M.J. Delwiche, and P.H. Brown. 2006a. Control of individual microsprinklers and fault detection strategies. *Precision Agriculture* 7: 85-99.

Coates, R.W., M.J. Delwiche, and P.H. Brown. 2006b. Design of a system for individual microsprinkler control. *Transactions of the ASABE* 49(6): 1963-1970.

Damas, M., A.M. Prados, F. Gómez, and G. Olivares. 2001. Hidrobus® system: Fieldbus for integrated management of extensive areas of irrigated land. *Microprocessors and Microsystems* 25(3): 177-184.

Gonda, L., and C.E. Cugnasa. 2006. A proposal of greenhouse control using wireless sensor networks. In *Proc. 4th World Congress Conference*, 229-233. St. Joseph, Mich.: ASABE.

Hebel, M.A. 2006. Meeting wide-area agricultural data acquisition and control challenges through zigbee wireless network technology. In *Proc. 4th World Congress Conference*, 234-239. St. Joseph, Mich.: ASABE.

Hebel, M.A., R. Tate, and D.G. Watson. 2007. Results of wireless sensor network transceiver testing for agricultural applications. ASAE Paper No. 073077. St. Joseph, Mich.: ASABE.

Jiang, X., J. Polastre, and D. Culler. 2005. Perpetual environmentally powered sensor networks. In *Fourth International Symposium on Information Processing in Sensor Networks*, 463-468.

Kim, Y., R.G. Evans, W. Iversen, and F.J. Pierce. 2006. Instrumentation and control for wireless sensor network for automated irrigation. ASABE Paper No. 061105. St. Joseph, Mich.: ASABE.

King, B.A., and D.C. Kincaid. 2004. A variable flow rate sprinkler for site-specific irrigation management. *Applied Engineering in Agriculture* 20(6): 765-770.

King, B.A., R.W. Wall, and L.R. Wall. 2005. Distributed control and data acquisition system for closed-loop site-specific irrigation management with center pivots. *Applied engineering in agriculture* 21(5): 871-878.

King, B.A., I.R. McCann, C.V. Eberlein, and J.C. Stark. 1999. Computer control system for spatially varied water and chemical application studies with continuous-move irrigation systems. *Computers and Electronics in Agriculture* 24(3): 177-194.

King, B.A., R.E. Reeder, R.W. Wall, and J.C. Stark. 2002. Comparison of site-specific and conventional uniform irrigation management for potatoes. ASAE Paper No. 022175. St. Joseph, Mich.: ASABE.

Koch, B., R. Khosla, W.M. Frasier, D.G. Westfall, and D. Inman. 2004. Economic feasibility of variable-rate nitrogen application utilizing site-specific management zones. *Agronomy Journal* 96: 1572-1580.

Lee, J.S., Y.W. Su, and C.C. Shen. 2007. A comparative study of wireless protocols: Bluetooth, UWB, ZigBee, and Wi-Fi. In *33rd Annual Conference of the IEEE Industrial Electronics Society*, 46-51.

Linden, D., and T.B. Reddy, eds. 2001. *Handbook of batteries*. 3rd ed. New York, N.Y.: McGraw-Hill.

Miranda, F.R., R.E. Yoder, J.B. Wilkerson, and L.O. Odhiambo. 2005. An autonomous controller for site-specific management of fixed irrigation systems. *Computers and Electronics in Agriculture* 48(3): 183-197.

Miranda, F.R. 2003. A site-specific irrigation control system. ASAE Paper No. 031129. St. Joseph, Mich.: ASABE.

NIST and USNO. 2008. The Official US Time. Boulder, Col.: National Institute of Standards and Technology, U. S. Naval Observatory. Available at: www.time.gov. Accessed 2 February 2008.

Suh, J., and M. Horton. 2004. Powering sensor networks. *Potentials, IEEE* 23(3): 35-38.

Torre-Neto, A., J.K. Schueller, and D.Z. Haman. 2000. Networked sensing and valve actuation for spatially-variable microsprinkler irrigation. ASAE Paper No. 001158. St. Joseph, Mich.: ASABE.

Verhagen, J. 1997. Site specific fertiliser application for potato production and effects on N-leaching using dynamic simulation modelling. *Agriculture, Ecosystems, and Environment* 66(2): 165-175.

Wang, D., T. Prato, Z. Qiu, N.R. Kitchen, and K.A. Sudduth. 2003. Economic and environmental evaluation of variable rate nitrogen and lime application for claypan soil fields. *Precision Agriculture* 4(1): 35-52.

Wang, N., N. Zhang, and M. Wang. 2006. Wireless sensors in agriculture and food industry--recent development and future perspective. *Computers and Electronics in Agriculture* 50(1): 1-14.

Warneke, B., M. Last, B. Liebowitz, and K.S.J. Pister. 2001. Smart dust: Communicating with a cubic-millimeter computer. *Computer* 34(1): 44-51.

Yang, C., J.H. Everitt, and J.M. Bradford. 2001. Comparisons of uniform and variable rate nitrogen and phosphorus fertilizer applications for grain sorghum. *Transactions of the ASAE* 44(2): 201-209.

Yiming, Z., Y. Xianglong, G. Xishan, Z. Mingang, and W. Liren. 2007. A design of greenhouse monitoring & control system based on ZigBee wireless sensor network. In *International Conference on Wireless Communications, Networking and Mobile Computing*, 2563-2567.

Zaman, Q.U., A.W. Schumann, and W.M. Miller. 2005. Variable rate nitrogen application in Florida citrus based on ultrasonically-sensed tree size. *Applied Engineering in Agriculture* 21(3): 331-335.