An Automated Meter Reading Application using Wireless Sensor Networks

Shoaib Azmat, David Johnson, Nate Klein, Matt Miller, Ahmad Najeeb, Josh Pelkey
Georgia Institute of Technology, School of Electrical & Computer Engineering
Sazmat3@gatech.edu, dej@gatech.edu, nate.klein@gatech.edu, mmiller@gatech.edu, ahmadka@gmail.com, jpelkey@gatech.edu

Abstract

Automating the task of meter reading can enable modern day utility companies to reduce costs, save man-hours, and eliminate needless errors. In the following paper two methods for wirelessly obtaining meter data are examined: (1) proximity drive-by approach where data is automatically transferred to a storage device when the utility vehicle is in close proximity to the meter, and (2) a fixed network scheme that relays meter data from house-to-house to a set collection point, eliminating the need to drive to individual houses.

1. Project Description

For gas and electric companies, meter reading is essential for business operation. Houses and apartments have utility meters that traditionally have required a utility company employee to travel out to each house on a periodic basis to "read" the meter. This task is ideal to turn over to a sensor network to relay the information back to a central locale to eliminate or reduce the need for a human meter reader. Of course, manual reading is relatively expensive when considering the employees' salaries, the vehicles to transport these employees, and the gallons of gas consumed to visit every meter, lest those laborers who have to endure the hardships of walking to each and every meter and record by hand the readings. Fortunately, with the proliferation of wireless technologies, meter reading can be revolutionized to cut costs.

Wireless sensor networks present a simple and cost-effective solution to meter reading for utility companies. By introducing a wireless sensor into meters at the customers' homes and the ability to network these sensors, the necessary information can be relayed back to a company without any of the costs of manual reading. Initial infrastructure costs for a wireless sensor network may seem unattractive when taking a shortsighted view. However, the cost of running the network post-installation is considerably smaller than continued manual reading. There is also a substantial difference between meter reading in urban and rural areas, which we will explore.

2. Challenges and Problems

The two different setups for our automated meter reading system presented very different design and implementation challenges. The first setting, in which a truck drives around and interrogates all sensors for their data, requires only point-to-point communication. The main issues encountered with this were: how to ensure that nodes only responded to the truck once and ensuring that all nodes were able to respond even if multiple nodes were within range of the truck at the same time. The second setting, in which data is relayed down a street, requires multi-hop communication. The main issues that we faced with this network were: ensuring that packets were transmitted to the sink reliably, preventing packets from looping infinitely in the network, and preventing a node's data from being recorded multiple times.

Single hop point-to-point communication does not require any sort of routing scheme, but deciding who has access to the channel, and for how long, is very important. We let the MAC protocol determine who can access the channel. If a node wins arbitration for the channel initially, it is allowed to complete the full transaction before losing access to the channel. Nodes that do not win arbitration for the channel simply time out, reset to the idle state, and wait for the next beacon to be sent from the sink. To prevent the same node from replying to multiple beacons, nodes turn off their wireless radio for a short amount of time after completing their transaction with the sink. This ensures that each node will only respond once and that the channel will be free for other nearby nodes to occupy.

Multi-hop communication presents difficulties in routing and reliability and can present problems with flooding the network. To ensure that every node's data reached the sink reliably, even in the presence of node blackouts/failures, we relied on a simple numbering scheme. Each node is assigned a unique number (must only be unique within that node's range), which could be identical to a house number. All nodes will listen to packets being sent and will automatically forward packets received from a node with a higher number. That number will then be inserted into a table of recently forwarded packets to prevent the same data from being forwarded more than once per node. The table is cleared periodically so that no data is blocked the next time that meter readings are sent (the next month). This ensures that data will always flow toward the sink, provides multiple routes for packets to traverse (resistance to node failures and loops), and prevents the network from being flooded with too many copies of the same data. The sink node also keeps a table of nodes from which it has received data. This will prevent the data from any one node from being recorded multiple times in one month (or whatever the period is).

3. SYSTEM ARCHITECTURE

Using sensors to automatically read and report accurate meter data is a general concept, which can be adapted and molded for different environmental and physical situations. Sensors can be deployed in different ways and configurations depending on what is needed for a given situation. For example, the sensor network deployed for meter reading for houses in a street can be very different from one meant for apartments in an apartment building. The fact is that there are many different environmental, physical and economic factors which influence the architecture of a sensor network, such as proximity between the sensors, the amount of RF interference, the medium through which power is provided, the method of collecting or reporting data, and so on. Based on what the type of input parameters given, there are primarily two main configuration categories, which are considered when deploying sensors for meter reading. These two categories are: The main Information Transfer Model to be used, and the Communication Medium to be used. Both will now be discussed below.

3.1. Information Transfer Models

There are primarily only two principle methods of retrieving data from sensors; to either collect data from sensors yourself (Pull-based), or to relay sensor data to the company offices (push-based). Both these approaches and their hybrids are discussed below:

3.2 Pull-Based Scheme

In the Pull-Based Scheme, data needs to be manually collected from the sensor-equipped meters. This is basically done by entering close proximity with the meter and using a data collection device to wirelessly retrieve data from the sensors. There are several different approaches to doing this too, such as 'driving by', in which a company vehicle simply drives by a range of houses, and the equipment installed in the vehicle automatically communicates and collects data from all the sensors it passes by. Sometimes a 'walk by' mode is used instead when 'drive by' is not performing well. The concept is the same as before, with the only change that since walking by is slower than driving by, data collection success rates would be better at the cost of more time to collect the same data.

3.3. Push-Based Scheme

In the Push-Based Scheme, meter-reading data is 'pushed' all the way to the company offices. This is done by creating a network out of the sensors deployed in an area, and then using this network to relay meter-reading data towards a common point, namely, the company offices. Needless to say, the obvious advantage of this approach is that no company employee would actually need to drive by or walk by a range of houses, since data will automatically arrive at the company offices. This would undoubtedly result in lower costs for the company, but like all things in life, it also has its disadvantages. The first and foremost issue with this Push-Based scheme is that sensors closer to the company offices would be spending a lot of their energy is relaying data across, and would thus require a relatively more heavy duty power supply. This issue needs to be addressed correctly in order to keep the overall scheme's performance positive. Otherwise, for example if the batteries of such sensor nodes need to be changed once every month anyway, then the plus point mentioned earlier that no employee physical presence is required would be cancelled out, since the employee would now have to come out to change the batteries. It's suggested that such nodes should be connected to a city power grid with some backup batteries installed directly into them. This would help keep these end nodes always on (ideally speaking!). Secondly, there is the issue of reliability. Suppose the node closest to the company offices stops working. How would data be relayed now? Because of this issue, it's required that there should be alternate paths available in the relay network too. All these issues boil down to the conclusion that this Push-Based scheme can only be used in very specific neighborhoods, where the houses are in close proximity with each other and where alternate relay paths are available too.

In Push-Based schemes, another thing that you have to consider is the topology you will be using to the relay network. In the previous paragraph, the 'Mesh' topology was assumed, in which sensors both transmit their own data, and also relay other sensors' data when needed. A practical real-world example of this topology is the Swedish city of Gothenburg, where the Mesh network is being used via the ZigBee protocol. The other common topology used for such relay networks is the Star Topology, in which all houses within a certain radius relay their data towards a centralized company office. The primary conceptual difference between Mesh and Star topologies is that, unlike the Mesh topology, relaying within a Star topology is normally not done via the sensor nodes themselves. Instead, additional equipment is installed at reachable places, which collect and relay data. These are normally Repeaters and Collectors.

3.4. Hybrid Pull/Push Schemes

As can be judged from above, both the Pull and Push Based Schemes have their own pros and cons. So why not find some middle ground between the two? This was the idea that led to the development of Hybrid Sensor Networks, which can toggle between the Push and Pull Schemes when the need arises. Sensors can be made to supply their meter data via both methods. This can be most costly, but the advantage of such a network is that it gives us redundancy, and allows the network to still be functional in case there are environmental problems. A hybrid Pull/Push Based Scheme can be advantageous for certain building types. For example, an apartment building can have a Push Based Scheme within the apartment building, which allows data to be passed downwards to the sensor on ground, and from there data can be collected using a Pull-Based mechanism. Such features of Hybrid Schemes have made them just as famous as the individual Pull and Push Based schemes.

3.5. Communication Medium

Whenever sensors have to be deployed for meter reading, one has to finalize on which communication medium to use for data transfer. There are many different communication channels available today which can be used, majority of which are briefly discussed below:

Pure RF: RF is today the most common and widely used communication medium for sensor networks. When used for the purpose at hand, communication can be done is one-way and two-way modes. In one-way, a sensor would only transmit its data after a specific interval, and would assume that someone has received its data. This assumption can surely prove to be incorrect in many cases, but for very high reliability sensor networks, assuming this is probably not wrong, as it can result in energy savings. Two-way communication mode is the common choice here as expected, in which sensors only communicate their data when there are asked for it. This does take somewhat more energy from the sensor to get the same data across, but its also helps us handle a lot communication errors. Hybrid One/Two communication modes are also used sometimes, in which the sensor can toggle when it detects some environmental change, such as, disconnection of main power, sensing RF interference, etc …

Wi-Fi: A Wi-Fi network can also be used to communicate sensor data. A good real-world example of this application is the city of Corpus Christi in Texas, US, which became the one of the first cities in the US to start a citywide Wi-Fi network for its inhabitants. Corpus Christi used this Wi-Fi backbone network to have all sensor readings transmitted through Wi-Fi. The advantage of using Wi-Fi over RF is obvious when the Push-Based Scheme is considered. Under such conditions, data would no longer be relayed to the company offices through sensors themselves, or repeaters/collectors. Instead, data can be directly sent to the company offices using the Wi-Fi network. This would result in all nodes experiencing the same energy consumption, and also would not require alternate relay paths. However, the disadvantage of using Wi-Fi for sending sensor data is reliability. Unlike RF, Wi-Fi networks can go down, and when this happens, alternate arrangements would need to be put in place. Also, transmitting/receiving data over a Wi-Fi network can consume a lot more energy than doing so over an RF channel. Thus, a Wi-Fi operating sensor network would need more powerful batteries.

Power Line Communication: Power Line Communication (PLC) is yet another way of communicating sensor data to the company offices. Communication takes place by utilizing the connection between a home's electric input and the electric company. Since there is an electric outlet coming to homes from the electric company, why not also use it for conveying sensor data?! This was the thought that led to the development of this idea. However, normally this method of conveying sensor data is only used in the case of electricity.

4. Implementation and Solutions

For our Meter Reading System, we implemented two different setups with solutions for both using the TinyOS platform and operating system. We tested our TinyOS code on multiple MicaZ Motes set up in various architectures to ensure our solution to each setup was working properly. For both solutions we used the serial forwarder and java printf statements to view sensing information furthering our ability to test our solutions for proper function. Our data was written both to the console and to a text file. This data would then need to be parsed and integrated into a main utility database were this to be carried forward to more realistic implementations.

Figure 1: The first implementation requires a vehicle to be within transmission range of each house to receive meter data

Our first implementation of a Meter Reading System operated under the assumption that a "meter reader" would drive through a neighborhood in a utility vehicle with a wireless sensor sink that would send out beacon requests in order to receive each and every utility meters' reading (see Figure 1). This scenario required point-to-point communication as stated previously and relies on highly trained utility personnel actively ensuring the proper recording of utility meter data. For this system we used a communication scheme where the master node, located in the utility truck, periodically sends out a beacon. Any sensor nodes within range that hear the beacon respond with a beacon reply. The utility truck will then send requests for data to sensors from whom a beacon reply was received. Once the remote sensors receive a request for data they send their current data, and then go to sleep after receiving an acknowledgement that the master node received the data. This whole process is shown visually in figure 2.

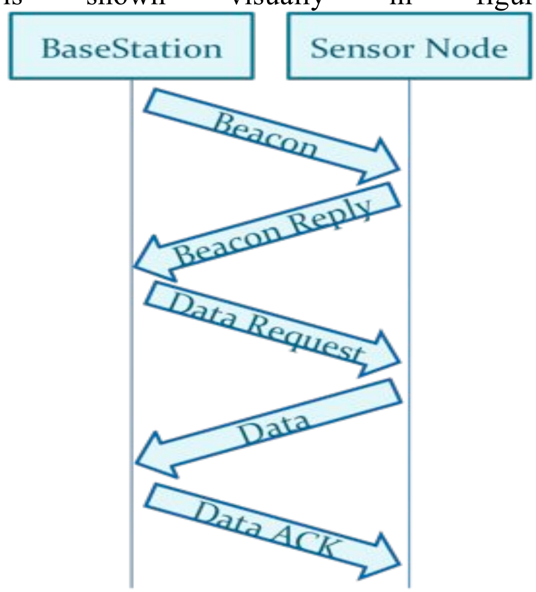

Figure 2: Communication sequence used in drive-by method

Our second implementation of a Meter Reading System built on the success of our first system, but eliminated the need for highly trained utility personnel driving around. Our solution harnessed the networking benefits of wireless sensors in a multi-hop communication network. This implementation's success was highly dependent on the engineer's node ID assignments, ensuring proper data flow, reliability and redundancy. The first scheme relied upon the master node sending out beacons and then requesting the data. In the second scheme we had remote nodes automatically send their data on a set interval. This eliminated the need for two-way communication simplifying the process and enabled us to get a good working prototype system ready for demonstration. In our fixed networks scheme each node periodically sends out it's own data and data from any higher numbered node is rebroadcasted by all lower numbered nodes that receive it. Thus choosing a solid numbering scheme for the network is essential in ensuring a proper tree structure with sufficient redundancy to get data from all sensors to the master node.

Figure 3: The fixed network scheme allows data to be forwarded from one node through the network to the base station

5. Evaluation

In this section we will evaluate the project outcome in the sense that we should be able to assess its applicability in real world scenarios. This project is in no way fit for commercial use and its purpose is to only serve as a proof of concept. It was demonstrated that sensor networks is an alternative for automatically reading the utility meters. For a real world product we need to add following basic enhancements to our project.

The first point to be considered is that we are using general-purpose sensor motes Micaz, which have a microprocessor board and a general sensor board with some basic sensing capabilities. We are using its light sensor to simulate the reading of a meter. In actuality we need an electro-mechanical sensor with some sort of electrical output. Figure 1 from [1] shows this where the mechanical sensor gives pulse output. So we have two options for the new system. We can either modify only the sensor board for this new sensing capability and integrate it with the Micaz processor board, or we come up with an entirely new sensor mote customized according to our application requirement. In the end, it really depends upon the cost analysis of the real world project.

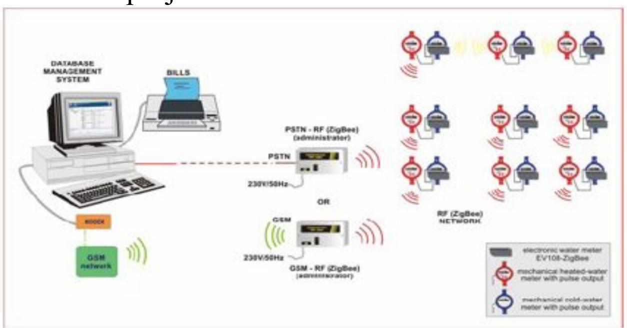

Figure 4: Wireless Automatic water reading (complete system based off ZigBee Technology)

The second thing we need to do is to deal with its error resiliency, fault tolerance, and end-to-end reliability. Both the solutions developed by us are basic in their nature showing the applicability of the concept. We should have some sort of error correction; like FEC, ARQ, or HARQ; depending upon the need of the application. We also need fault tolerance specifically in the second case of forwarding data via multiple hops, so that if a node dies there should be alternative for routing the packets. In other words the routing algorithm should have redundant paths for forwarding packets as explained in [2]. And lastly, since we were working in limited scope with only few motes in a lab, end-to-end reliability and congestion control was not an issue. So, we have not come up with some transport layer protocol. In large network with higher traffic and delays these things become very important, so we need to develop some transport layer protocol depending upon the needs of the application from the various options available.

The third thing we need is the active involvement of the base station. In the present case the base station is acting almost as a passive element, which issues beacons and saves the data it receives. In the real world scenario, the base station should have clear view of the network topology so that it can see occurrence of faults, e.g. a mote not working. Also it should be able to plot the data and do basic analysis and actions based on it, depending on the actual application needs. In short we need some sort of data management software at the base station, as shown in Figure 4.

Last but not least, we also need to discuss the network topology for real world scenario. The details for this depend on the requirements for a particular situation. The things we need to consider is that how many tiers should be in the topology? It depends on whether the sink communicates directly with each mote or would there be other tiers in between like the concept of cluster head. Is the sink mobile or static? Meaning do we need to send data via multiple hops to sink or is data is collected from each sensor mote. How are we connected to utility centre? The situations here can be such that we have multiple data collection centers i.e. sinks at various sites and each of them is acting as a gateway to connect to the Internet via PSTN or GSM, as shown in Figure 4 and also in Figure 5 below. Figure 5, as in [3], is a commercial topology as used by Nanotron Technologies for automatic meter reading. So for this we need to develop a gateway for our system to talk to TCP/IP. In this way base station at utility center can get access to the network via Internet. In short, precise answers to all the questions posed above depend on the actual scenario in which we are going to deploy our network.

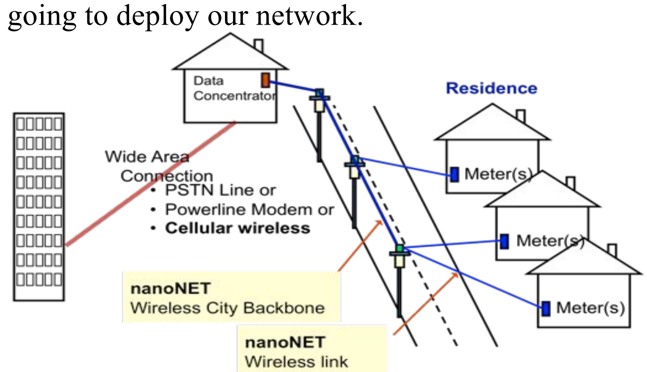

Figure 5: Automatic Meter reading (Nanotron Technologies)

6. References

[1] Gordan Štruklec, Vedran Bilas. Wireless Automatic Water-meter Reading System, 2006-07

[2] Kwang-il Hwang. Fault-tolerant ZigBee-based Automatic Meter Reading Infrastructure, 2009

[3] John V. Lampe. Presentation on CSS product examples, Nanotron Technologies GmbH, 2007

[4] "Automating Your Utility Meter Reading Process – FAQ" [Online], Access April 2010. Available HTTP: http://www.avanticompany.com/articles/amr.html