*Proceedings of the 2001 Winter Simulation Conference*
*B. A. Peters, J. S. Smith, D. J. Medeiros, and M. W. Rohrer, eds.*

# SIMULATING NETWORKS OF WIRELESS SENSORS

Sung Park
Andreas Savvides
Mani B. Srivastava

Networked Emebedded Systems Laboratory
Electrical Engineering Departments
University of California, Los Angeles
Los Angeles, CA 90095, U.S.A.

## ABSTRACT

Recent advances in low-power embedded processors, radios, and micro-mechanical systems (MEMs) have made possible the development of networks of wirelessly interconnected sensors. With their focus on applications requiring tight coupling with the physical world, as opposed to the personal communication focus of conventional wireless networks, these wireless sensor networks pose significantly different design, implementation, and deployment challenges. In this paper, we present a set of models and techniques that are embodied in a simulation tool for modeling wireless sensor networks. Our work builds up on the infrastructure provided by the widely used ns-2 simulator, and adds a suite of new features and techniques that are specific to wireless sensor networks. These features introduce the notion of a sensing channel through which sensors detect targets, and provide detailed models for evaluating energy consumption and battery lifetime.

## 1 INTRODUCTION

The marriage of ever tinier and cheaper embedded processors and wireless interfaces with micro-sensors based on micro-mechanical systems (MEMS) technology has led to the emergence of ***wireless sensor networks*** as a novel class of networked embedded systems. Many interesting and diverse applications for these systems are currently being explored. In indoor settings, sensor networks are already being used for condition-based maintenance of complex equipment in factories. In outdoor environments, these networks can monitor natural habitats, remote ecosystems, endangered species, forest fires, and disaster sites.

The primary interest in wireless sensor networks is due to their ability to monitor the physical environment through ad-hoc deployment of numerous tiny, intelligent, wirelessly networked sensor nodes. Because of the large numbers of sensor nodes required, and the type of applications sensor networks are expected to support, sensor nodes should be small, tetherless, and low cost. Due to these requirements, networked sensors are very constrained in terms of energy, computation and communication. The small form factor requirement prohibits the use of large long lasting batteries. Low production costs and low energy requirements suggest the use of small, low power processors, and small radios with limited bandwidth and transmission ranges. The ad-hoc deployment of sensor nodes implies that the nodes are expected to perform sensing and communication with no continual maintenance and battery replenishment. The energy constraints call for power awareness, which in turn leads to additional tradeoffs. The high-energy costs associated with wireless transmission, made particularly severe for sensor networks because nodes with small antenna heights placed on the ground see $1/r^4$ wireless link path loss coupled with the ever reducing cost of processing has led to a the adoption of a distributed computing viewpoint for wireless sensor networks. Instead of simply sending the raw data (perhaps compressed) to a gateway node, in typical applications the nodes in wireless sensor networks perform computation for decision making within the network, either individually via techniques such as signature analysis or in local clusters using coherent combining of raw sensor signals (i.e. beam forming) or non-coherent combining of decisions (i.e. Bayesian data fusion). By performing the computation inside the network, communication may be reduced thus prolonging the network lifetime

We construct a versatile environment in which sensor networks can be studied. This environment employs a wide range of models to orchestrate and simulate realistic scenarios. Furthermore, since power consumption is also a key design factor, we emphasize power consumption and battery behavior models. First we create a set of sensor node models that are derived from the empirical power characterization of two different nodes representing two extremes; the WINS node (Rockwell Scientific Company LLC. 2001) from Rockwell Science Center and the Medusa node, an experi-

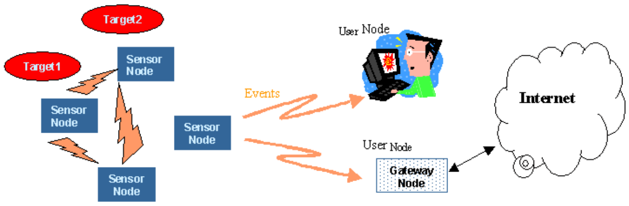

Figure 1: Sensor Network Scenario

mental prototype that we have constructed. These sensor node models are combined into the widely used event queue based network simulator, ns-2 (ns-2 Simulator 2001). By introducing the notion of *sensing channels* in our simulation environment and a flexible and highly parametrizable scenario generation tool, we can study the power consumption of sensor nodes by instrumenting complex sensor network scenarios in a detailed graphical environment.

## 2 RELATED WORK

Although sensor networks have recently received a lot of attention, there are still not many formal tools available for the systematic study of sensor networks. The work in (Ulmer 2001) presents a Java based simulator for sensor networks. This is an online simulator that can create and simulate simple topologies but does not have any explicit models for sensors or power management. Up to this point there is no publication on this work. On the network simulation, numerous simulators are currently available such as GloMoSim, OPNET and ns-2. These simulators provide great flexibility in the simulation of wireless ad-hoc networks at all layers. Despite their effectiveness, these tools are currently not equipped for capturing all the aspects of interest in sensor networks.

## 3 SIMULATION ARCHITECTURE OVERVIEW

We motivate our discussion with an example of a sensor network illustrated in figure 1. In this example, a set of wireless nodes equipped different sensor for monitoring natural habitat. The results of these sensors are processed within the network and the final sensing report is forwarded via wireless links to the gateway nodes that makes the results available on the internet. The main goal of our work is to recreate such scenarios in a versatile simulation environment where the behavior of the sensor network can be analyzed.

In our simulation environment, a typical sensor network scenario will consist of three types of nodes: 1) ***sensor nodes*** that monitor their immediate environment, 2) ***target nodes*** that generate the various sensor stimuli that are received by multiple sensor nodes via potentially many different transducers (e.g. seismic, acoustic, infrared) over different sensor channels; e.g. a moving vehicle generates ground vibrations that trigger seismic sensors and sound that triggers acoustic sensors, and 3*)* ***user nodes*** that represent clients and administrators of the sensor network. Shown in figure 2, three type of node models make up the key building blocks of our simulation environment. The sensor nodes are the key active elements, and form our focus in this section. In our model, each sensor node is equipped with one wireless network protocol stack and one or more sensor stacks corresponding to different types of transducers that a single sensor node may possess. The role of the sensor protocol stacks is to detect and process sensor stimuli on the sensing channel and forward them to the application layer which will process them and eventually transmit them to a user node in the form of sensor reports. In addition to the protocol and sensor stacks that constitute the algorithmic components, each node is also equipped with a power model corresponding to the underlying energy-producing and energy-consuming hardware components. This model is composed of an energy provider (the battery) and a set of energy consumers (CPU, Radio, Sensors). The energy consuming hardware components can each be in one of several different states or modes, with each mode corresponding to a different point in performance and power space. For example, the radio may be in sleep mode, receive mode, or one of several different

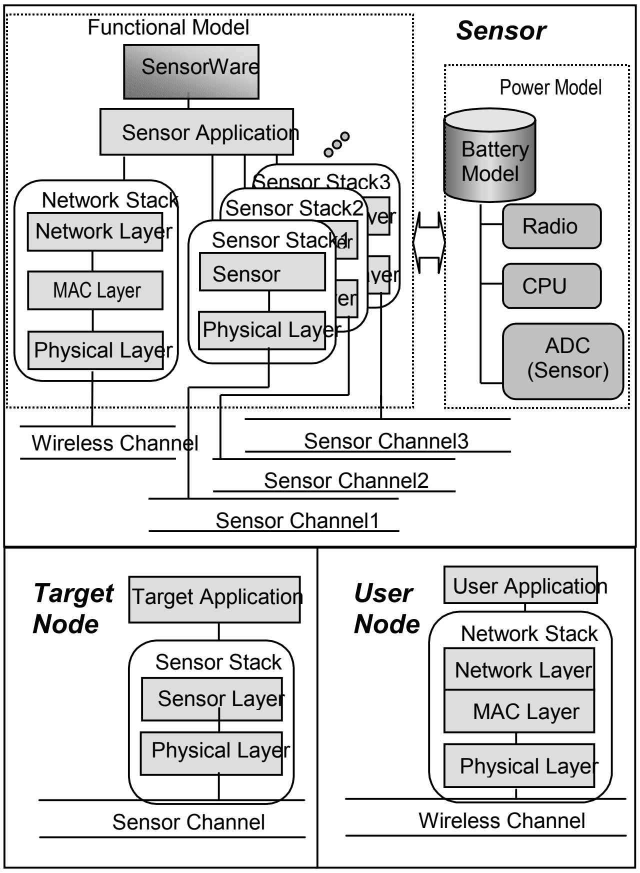

Figure 2: Sensor Node Model Architecture

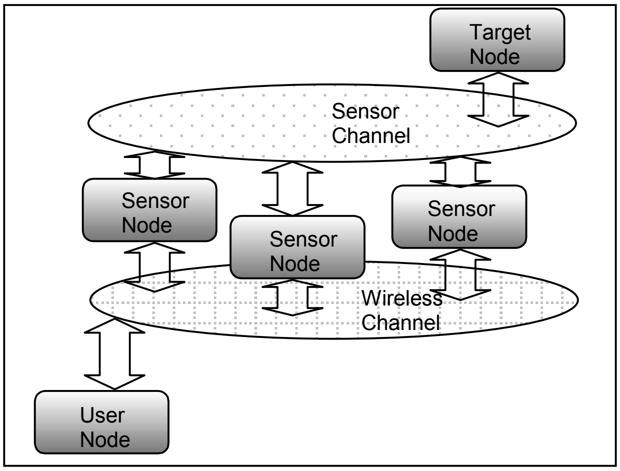

Figure 3: Sensor Network Model Architecture

transmit modes corresponding to different symbol rates, modulation schemes, and transmit power. Similarly, the CPU may be in sleep mode, or one of several different active modes corresponding to different frequency and voltage. The algorithms in the network and sensor stack control the change in mode of the power consumers. For example, the MAC protocol may change the radio mode from sleep to receive. In return, the performance of the algorithms may depend on the mode. For example, the time taken by the physical layer in the network protocol stack would depend on the data rate of the mode the radio is currently in. All of this is accomplished by having the algorithms in the network and sensor stacks issue mode change events to the power consumer entities, and having the algorithms read relevant parameter values from those entities. Algorithm-induced changes in the operating modes of power consuming hardware entities in turn affect the current drawn by them from the battery which delivers the power corresponding to the sum of current (or power) drawn by each power consumer. Internally, the battery entity depletes its stored chemical energy according to the efficiency dictated by the battery model.

Figure 3 illustrates how a typical sensor network will be constructed and simulated using our simulation environment. In figure 3, the wireless channel and sensor channel form separate communication mechanisms where events from different nodes are passed through. A typical scenario will involve a target node passing through a group of sensor nodes deployed in the field. As the target node moves around, it gives out sensor signal in the form of events through the sensor channel and each sensor node detects the events based on propagation model implemented in each node's sensor stack. When sensor nodes determine the sensor signals (events) are noteworthy, they transmit packets (also in the form of events) through the wireless channel destined to the user node.

By separating the sensor channel and the wireless channel, our sensor network model makes easier to simulate and analyze the operation of sensor network where the sensor signal detection events and wireless communication events can be received or transmitted concurrently. Moreover, by allowing sensor node to connect to multiple sensor channels, our simulation environment provides ability to analyze complex behaviors of sensor nodes' reaction to multiple sensor signals (i.e. seismic vibration, sounds, temperature, etc..) that can be detected all at the same time. In the following section, we discuss each components of the sensor node model shown in figure 2, and explain how we construct the model of different sensor nodes' components.

## 4 FRAMEWORK OF SENSOR NETWORK SIMULATION

### 4.1 Node Placement and Traffic Generation

In studying the performance of a wireless sensor network for a given application, a crucial element is the overall deployment scenario which includes the node placement topology, the radio ranges, the sensing ranges, the trajectories of the targets and resultant event traffics, and the trajectories of the user nodes and their query traffics. All these elements contribute to the different design trade-offs that can be made, and it is crucial to evaluate the effects of a new algorithm or protocol under diverse deployment scenarios.

To study such effects, we have developed a detailed scenario generation and visualization tool that enables us to construct detailed topologies and sensor network traffic. Our simulation environment enables us to assess the requirements of a sensor network under different circumstances by generating detailed scenario input to our simulations. This complements the scenario generation techniques provided in (ns-2 Simulator 2001) which are mainly targeted to ad-hoc wireless communication networks. Sensor node placement can vary depending on intended the task on the network. For example, to monitor wildlife in a forest, sensors may be uniformly distributed in the forest. If however, the sensor network is deployed for perimeter defense, then the sensors will most likely be distributed around a specified perimeter in a two dimensional gaussian distribution. In some other cases, the sensors may be manually placed according to the requirements of the user.

Besides placement, the traffic requirements may be even more diverse. Sensor network traffic can be classified into 3 main types: 1) *user-to-sensor* traffic, which is a result of user commands and queries to the network, 2) *sensor-to-user* traffic, which consists of the sensor reports to the user and 3) *sensor-to-sensor* traffic, which includes collaborative signal-processing of sensor events in the network before they are reported to the user. The last type of traffic is the most complex, and it depends on the sensing method.

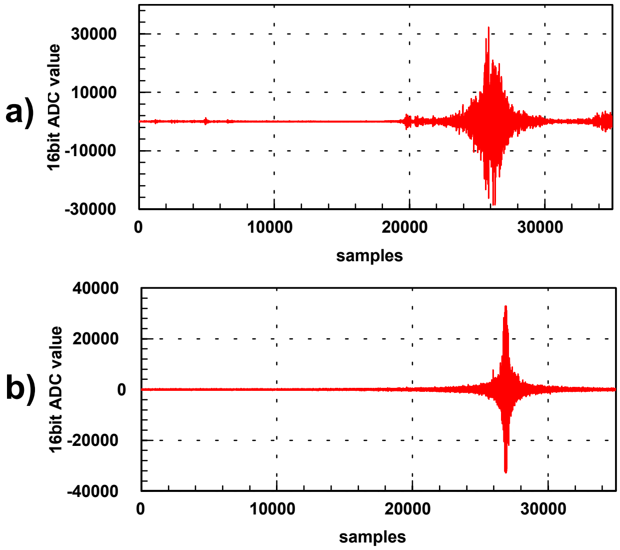

Figure 4: a) Real Target Seismic Signature b) Simulated Target Seismic Signature

## 4.2 Sensor Stack and Sensor Channel

The *sensor stack* simulates how a sensor node generates, detects and processes sensor signals. In sensor node model (figure 2), the *sensor stack* is a signal sink that is responsible for triggering the application layer every time a sensing event occurs. Various trigger functions ranging from simple sensing schemes to elaborate signal processing functions can be implemented in the sensor stack. In target node model, the *sensor stack* acts as a signal source. The sensor stack of a target node will contain a signature that is unique to the type of target the target node is modeling. The signature is then transmitted through various mediums (ground, air, free space, water, etc..) as the target node moves around. figure 4a and 4b show a real and a simulated signature obtained from a seismic sensor triggered by ground vibration from a traveling vehicle.

In figure 4, the ground is the medium that transmits the vibrations to the seismic sensor. We refer to this medium as the sensor channel, a model of a medium which sensor events such as seismic vibration, sounds, or infrared signals are traveled through. The type of medium can differ based on the type of sensor being modeled (seismic, acoustic, infra red, ultrasonic). Moreover, depending on the medium being modeled, the propagation of signal can differ. For instance, a sound moving through the air will have different propagation as the same sound moving through the water. In order to incorporate all these different aspects of the sensor network in to our simulation, we implement a simple sensor stack and sensor channel model by modeling the target node as a gaussian source whose signal amplitude is modeled as a gaussian random variable with the mean equal to zero and the variance $\sigma^2$. As the target travels through the sensor network, the target exerts the vibration signals (signal events) into the sensor channel periodically. The sensor channel then delivers this events to each sensor node's sensor stack and each sensor node adjusts the signal strength of the target based on sensor channels propagation model. The figure 4b demonstrates the signal strength variation as a target passes by a sensor node on a straight line. As the target approaches the sensor node, the signal strength increases, and as the target moves away, the signal attenuates rapidly. In this simulation the sensor signal was attenuated at a rate of 1/r where r is the distance between the target and the sensor.

## 4.3 Hardware Components Characterization

$$I_{sensor} = \frac{V_{test}}{R_{test}}$$

$$V_{Sensor} = V_{supply} - V_{test}$$

$$Power = I_{sensor} \times V_{sensor}$$

$$Energy = Power \times time$$

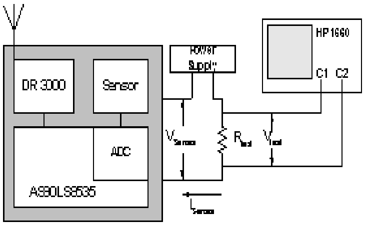

Figure 5: Power Measurement Configuration

Table 1: Experimental Node Current Consumption

| Mode ID | CPU | Radio(OOK Modulation) | ADC | Total Current |
|---|---|---|---|---|
| 1 | Active 2.9mA | Tx-19.2kbps 5.2mA | On | 8.1mA |
| 2 | Active 2.9mA | Tx-2.4kpbs 3.1mA | On | 6.0mA |
| 3 | Active 2.9mA | Rx:4.1mA | On | 7.0mA |
| 4 | Sleep 1.9mA | Sleep:5μA | On | 1.9mA |
| 5 | Off 1μA | Sleep:5μA | Off | 6μA |

We construct our power models by performing measurements of the hardware power consumption using an HP 1660 oscilloscope, a bench power supply, and a high precision resistor. The measurement setup and power relationships are shown in figure 5. By characterizing each component of the sensor nodes we enable the simulated nodes to operate at different modes in which the power management schemes can switch different components on and off. Using the configuration in figure 5, the total current consumption of our experimental sensor node is obtained in Table 1. The measurements listed in Table 1 provide a better insight into the power consumption of the sensor nodes since the actual power consumption is oftentimes different from the typical values provided in the manufacturer data sheets depending on the mode of operation.

### 4.4 Battery Models

The Battery Model simulates the capacity and the lifetime of the sole energy source of the sensor node, the battery. In reality, battery behavior highly depends on the constituent materials and modeling this behavior is a difficult task. Although the battery can be viewed as a energy storage, the main goal of the sensor network is to increase the lifetime of the battery. Thus, in this section, we focus on how battery's capacity can be modeled based on the energy consumers' behavior. We propose 3 different types of battery models to study how different aspects of real battery behavior can affect the energy efficiency of different applications. The metrics that are used to indicate the maximum capacity of the battery is in the unit of Ah (Ampere*Hour). The metric is a common method used by the battery manufacturers to specify the theoretic total capacity of the battery. Knowing the current discharge of the battery and the total capacity in Ah, one can compute the theoretical lifetime of the battery using the equation , $T = \frac{C}{I}$ , where *T*=battery lifetime, *C*=rated maximum battery capacity in Ah, and *I*=discharge current.

#### 4.4.1 Linear Model

In Linear Model, the battery is treated as linear storage of current. The maximum capacity of the battery is achieved regardless of what the discharge rate is. The simple battery model allows user to see the efficiency of the user's application by providing how much capacity is consumed by the user. The remaining capacity after operation duration of time $t_d$ can be expressed by the following equation.

$$\text{Remaining capacity (in Ah)} = C = C' - \int_{t=t_0}^{t_0+t_d} I(t)\,dt \qquad (1)$$

where *C'* is the previous capacity and *I(t)* is the instantaneous current consumed by the circuit at time *t*. Linear Model assumes that *I(t)* will stay the same for the duration $t_d$, if the operation mode of the circuit does not change ( i.e. radio switching from receiving to transmit, CPU switching from active to idle, etc.. ) for the duration $t_d$. With these assumptions equation 1 simply becomes as the following.

$$C = C' - \int_{t=t_0}^{t_0+t_d} I(t)dt = C' - I \cdot t\Big|_{t_0}^{t_0+t_d} = C' - I \cdot t_d \qquad (2)$$

The total remaining capacity is computed whenever the discharge rate of the circuit changes.

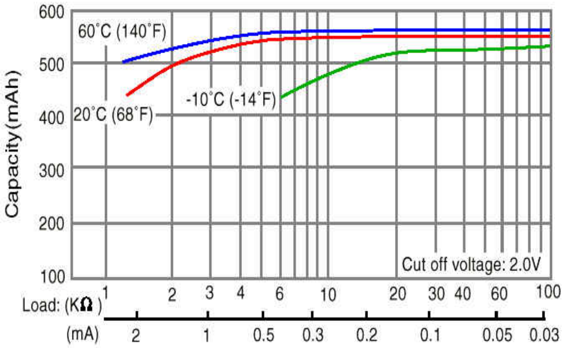

Figure 6: Capacity vs. Discharge Rate Curve for CR2354 (Matsushita Electric Corp. of America 2001)

#### 4.4.2 Discharge Rate Dependent Model

While Linear Model assumes that the maximum capacity of the battery is unaffected by the discharge rate, Discharge Rate Dependent Model considers the effect of battery discharge rate on the maximum battery capacity. In [15] [16], it is shown that battery's capacity is reduced as the discharge rate increases. In order to consider the effect of discharge rate dependency, we introduce factor *k* which is the battery capacity efficiency factor that is determined by the discharge rate. The definition of *k* is, $k = \frac{C_{eff}}{C_{\max}}$ , where $C_{eff}$ is the effective battery capacity and $C_{max}$ is the maximum capacity of the battery with both terms expressed in unit of Ah. In Discharge Rate Dependent Model, the equation 1 is then transformed to the following.

$$C = k \cdot C' - I \cdot t_d \qquad (3)$$

The efficiency factor *k* varies with the current *I* and is close to one when discharge rate is low, but approaches 0 when the discharge rate becomes high. One way to find out corresponding *k* value is for different current value of *I* is to use the table driven method introduced in (Simunic 1999).

#### 4.4.3 Relaxation Model

Real-life batteries exhibit a general phenomenon called "relaxation" explained in (Fuller 1994, Linden 1995, Chiasserini 1999). When the battery is discharged at high rate, the diffusion rate of the active ingredients through the electrolyte and electrode falls behind. If the high discharge rate is sustained, the battery reaches its end of life even though there are active materials still available. However, if the discharge current from the battery is cutoff or reduced during the discharge, the diffusion and transport rate of active materials catches up with the depletion of the materials. This phenomenon is called relaxation effect, and it gives the battery chance to recover the capacity lost at high discharge rate. For a realistic battery simulation, it's important to look at the effects of relaxation as it has effect of lengthening the lifetime of the battery. For our simulation, we adapt the analytical model introduced in (Fuller 1994) which takes discharge rate as input and computes the battery voltage over the simulation duration.

## 5 EXAMPLE STUDY CASE

In this section we demonstrate some of the main capabilities of our tool by studying the performance of different battery models with various sensor node operation profile.

### 5.1 Low Rate/Low Power vs. High Rate/High Power

In this case study, we evaluate the battery consumption of our experimental sensor node by considering different operation profiles. In section 4.3, we have discussed how each component of our sensor node has different power consumption depending on its operation mode. In this section, we examine how the combination of the operation modes of different components affects the aggregate power consumption of the sensor node. The scenario involves two sensor nodes (a transmitter and a receiver) that are within the transmission range of each other (approximately 15 meters apart) where the transmitter needs to transmit a 2MB file to the receiver. For the purposes of our discussion we define 5 different operation modes for our experimental node shown in table 1. To examine the energy consumption and communication tradeoffs we evaluate 3 different data transmission policies.

1)19.2 kbps continuous transmission: The transmitter sends data at the highest data rate without any break. The transmitter will be operating in mode 1 and the receiver will be operating in mode 3; 2) 2.4 kbps continuous transmission: With lower data rate the sender can transmit at a lower power level to reach the receiver. The transmitter will be operating in mode 2 and the receiver will be operating in mode 3; 3) 19.2 kbps pulse transmission: The transmitter sends data intermittently at the highest power level. While the transmitter is not transmitting, the transmitter puts the CPU and Radio to sleep. The transmitter power cycle its component by transmitting one 60 byte packet at 19.2 kbps for .025 sec and sleeps for .125 sec until all the data is received by the receiver. The transmitter will be switching between modes 1 and 4, and the receiver will be switching between mode 3 and 4.

Figure 7a shows the effect on each battery model capacity after the 2 MB data transfer for the three transmission methods described above. This experiment was performed for all three battery models described in section 4.4. Initially, all batteries were set to a capacity level of 10 mA*hour. The left half of figure 7a describes the remaining battery capacity of the transmitter after the file transfer, and the right half shows the receiver battery capacity. The solid bar in the figure indicates the total time for data transfer. Looking at the solid bar, it is clear that the sending the file at high data rate takes the least time thus the least battery capacity. Although the 2.4 kbps transmission and 19.2 kbps transmission took the same amount of time to transmit the data, 19.2 kbps pulse transmission saved much battery capacity due to the sleep period. Figure 7a also shows how different battery models exhibit different characteristics under different transmission methods. The linear model shows how optimum battery will behave as it shows the theoretical capacity of the battery under any discharge current. On the other hand, the rate dependent model accurately describes how real batteries will behave when there is a constant discharge for long duration. This is shown in 2.4 kbps transmission where the remaining capacity of rate dependent model is substantially less than the

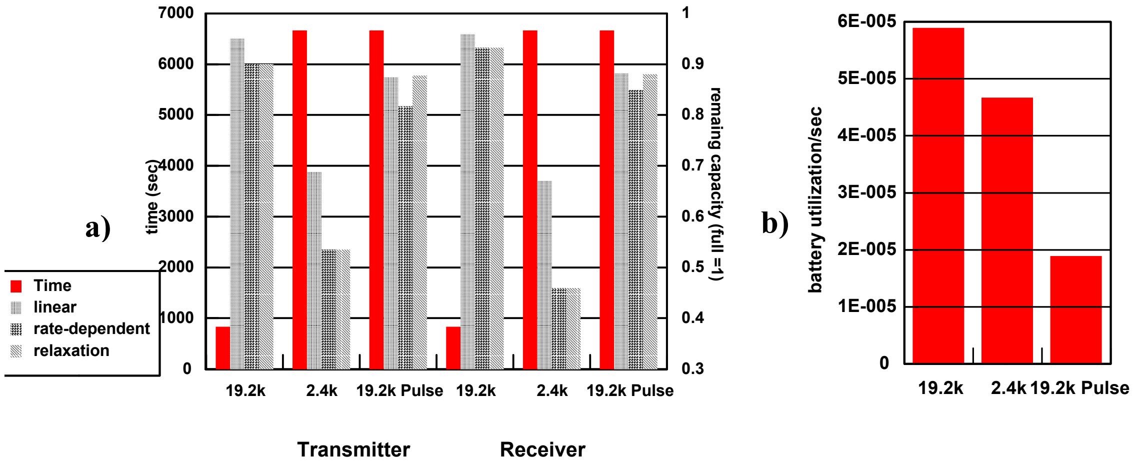

Figure 7: a) Battery Capacity Usage Under Different Discharge Profiles, b) Battery Utilization Rate

linear model. The other interesting model is the relaxation model which exhibits the both discharge rate dependent capacity and recovery effect. Since the relaxation model has recovery properties, the difference between relaxation model and rate dependent model is shown in the pulsed transmission cases. In figure 7a the relaxation model has the same remaining battery life as rate dependent model for 19.2kbps and 2.4 kbps continuous transmission and reception. However, in 19.2 kbps pulse transmission and reception, the relaxation model has almost equal capacity as the linear model due to the capacity recovery during the sleep mode.

### 5.2 Monitoring a Moving Vehicle in a Sensor Field

In this implementation we first show the effect of traffic on the sensing and communication traffic and then we evaluate simple power management scheme using the same sensor node setup as in the previous subsection. For this we have implemented a lightweight protocol stack similar to what one would expect to have on a tiny sensor node. The radio transmission and reception are driven by a TDMA based medium access control (MAC) protocol based on unique slot assignment algorithm derived from [9]. The MAC protocol assigns a unique slot to each node over a 2-hop radius and each node is aware of its one-hop neighbors and their corresponding slot assignment. For routing, we have implemented a very lightweight table-driven routing protocol with table size of one (next hop to user node). The motivation for TDMA scheme comes from our result in section 5.1 where a pulse transmission and reception can improve the battery utilization in the long term. In our power aware TDMA scheme this requirement is met for both the transmission and reception of packets. For transmission, a node is only allowed to transmit in its assigned time-slot. For reception, a node only needs to listen to the wireless channel for the duration of the slots that are already assigned to its one-hop neighbors. For the purposes of our discussion we refer to all the other remaining slots as *idle slots*.

In this scenario, a small cluster of 10 sensors equipped with seismic sensors is deployed to detect a bypassing truck as shown in figure 8a. The seismic sensors run at a sample rate of 400Hz to produce 16 bit samples. The sensor nodes are configured to report back to a gateway node that makes the results available on the Internet. Each sensor is programmed to transmit a report to the gateway within 5 seconds from the moment the ground vibrations from the truck are detected. If at least 2048 samples are obtained, the node can perform coherent detection and it will transmit a 10 byte to report the target type. This short packet is called “coherent traffic”. If however, the node does not have enough samples at the end of the 5-second period, it enters a non-coherent detection mode and transmits all its available samples to the gateway node which performs sensor data combination (called “beamforming” [20]) to improve the detection accuracy. Since the sensor node transmits raw data when it enters non-coherent detection mode, the size of non-coherent data tends to be lot larger than the coherent traffic. This non coherent raw data is referred to as “non-coherent traffic” During the simulation, the network traffic will be consist of coherent and non-coherent traffic depending on whether the individual sensors successfully classified the target. We discuss the result of the simulation in the following two sub sections.

#### 5.2.1 Efficiency of Power Management Scheme with TDMA

One apparent advantage of TDMA over other CSMA random access MAC protocols is the fact that the sensor nodes do not have to be in receive mode during the time slots where none of its neighbors are schedule to transmit. This allows the sensor nodes to perform a simple power management scheme that puts the CPU and radio to sleep during *idle slots* to conserve battery capacity. With this setup, we evaluate the efficiency of battery capacity utilization when this simple power management scheme is used. Figure 8a is the scenario used for our evaluation. It consists of 100 nodes uniformly distributed across a sensor field. The

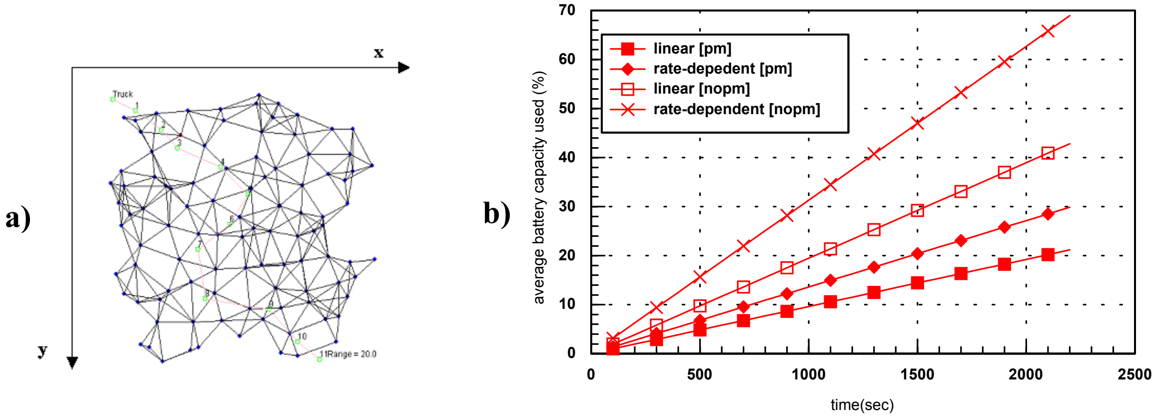

Figure 8: a) 100 Node Test Topology, b) Battery Capacity Usage

target travels at approximately 22 mph (10 m/s) through the track every 2 minutes. The target signals have an effective range of 20 meters. As the target travels through the sensor field, every node within the range of the target start collecting signal samples at 400 Hz, then send reports to the user node. We tested this scenario using the linear model and the rate dependent model by looking at battery utilization when the power management scheme is implemented (PM) and when there is no power management (NOPM).

The current drawn by each node will be similar to the cases described in section 5.1 with power management case resembling the 19.2 kbps pulse transmission and the no power management case resembling the 19.2 kbps continuous transmission. Figure 8b shows the average battery capacity utilization for each node. The bottom two curves show the difference in battery capacity utilization when the power management was used and the top two describe the cases when no power management is used. As the figure indicates, there is almost 100% improvement of battery utilization with the power management.

### 5.2.2 Effect of Sensor Power Cycle

In addition to the battery saving achieved by the TDMA power management scheme, we further look at how the sensor nodes can power cycle their sensors to conserve battery capacity. In this scenario (figure 9a), a square grid of sensor network is strategically placed over a flat field. The target travels along a pre-specified path and the sensor nodes attempts to make either coherent or non-coherent detection as described in the previous section. One difference in this scenario is that the sensor nodes attempt to turn on the sensors only intermittently to conserve power. When the sensor is turned off, the CPU of our experimental sensor node can go to mode 5 (table 1) where the power consumption is in the range of microwatts. However, the trade-off comes from the reduction of detection and classification accuracy since the sensor will miss the sensor signals coming from the target when they are turned off. The cost of such missed events may be very application specific. If the target occurrence is very frequent, it may be okay to miss its detection, but if the occurrence is very infrequent, it may be very crucial to detect that one incidence. It is possible that the whole sensor network may have been deployed to detect that “one” incidence. Therefore, in designing sensor network it’s crucial to look at the application requirement as well as the target characteristics to guarantee of certain quality of service (QoS) similar to the one provided in telecommunication network. One such QoS guarantee will be something like “a target with a 20 mph speed following this track will not pass through the sensor field undetected”. In this section, we try to look at what would be the maximum battery power saving that can be achieved while providing such QoS guarantees.

We look at the impact of a simple power management scheme which randomly wakes up the sensor within a prespecified time window of 100 seconds and stay up for different percentage of duration. Figure 9b shows the battery capacity used and the amount of coherent data bytes transmitted for different power cycle durations. The plot indicates that there is a rapid decrease in coherent detection as the power cycle percentage decrease from 60% to 50%. On the other hand, the battery utilization steadily decreases as the power cycle percentage decreases.

## 6 CONCLUSIONS

We have demonstrated a flexible toolset for studying power consumption in sensor networks. With the flexible architecture that closely simulate the behavior of real sensor network, accurate power models of sensor nodes and

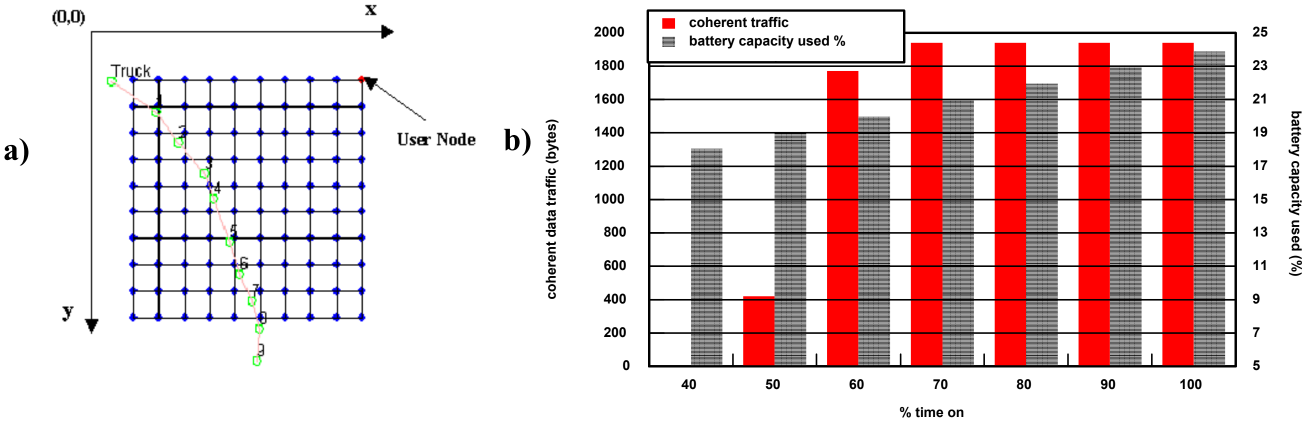

Figure 9: a) Sensor Scenario in Grid Sensor Network b) Coherent Traffic Data and the Battery Efficiency Of Various Power Cycle Durations

analysis of battery behavior are utilized in a tool to evaluate power consumption in the context of a realistic scenario. With these results we can assess the power consumption for new sensor nodes that are currently under development. Furthermore, this tool has been an indispensable aid in estimating the resources required for the network protocols to function correctly in new node architectures. By simulating and validating target protocols we can also get a good indication of code size and memory requirements thus resulting in feasible low cost designs. We envision that this set of tools will play an instrumental role in the design and implementation of new application specific sensor networks.

## REFERENCE

Chiasserini, C. F., and R. R. Rao. 1999. Pulsed battery discharge in communication devices. In *Proceedings of Mobicom 99*, Seattle, August 1999.

Fuller, T. F., M. Doyle, and J. Newman. 1994. Simulation and Optimization of the Dual Lithium Ion Insertion Cell. *Journal of Electrochem. Soc*., vol. 141, no. 4, pp 1-10.

Linden, H.D. 1995. *Handbook of Batteries*. 2nd ed. New York: McGrawHill.

Matsushita Electric Corp. of America. 2001. *Panasonic lithium coin cell battery datasheet*. Available via `http://www.panasonic.com/industrial_oem/battery/battery_oem/chem/lith/lith.htm` [accessed July 9, 2001].

ns-2 simulator. 2001. *ns-2 Simulator*, Available via `http://www.isi.edu/nsnam/ns/` [accessed July 9, 2001].

Rockwell Scientific Company LLC. 2001. *WINS (Wireless Integrated Network Systems) Project*. Available via `http://wins.rsc.rockwell.com/` [accessed July 9, 2001].

Simunic, T., L. Benini, and G. De Micheli. 1999. Energy-Efficient Design of Battery-Powered Embedded Systems. In *Proceedings of International Symposium on Low Power Electronics and Design*, 212-217, Piscataway, New Jersey.

Ulmer, C. 2001. *Sensor Network Simulator*, Available via `http://users.ece.gatech.edu/~grimace/research/sensorsimii/` [accessed July 9, 2001].

## AUTHOR BIOGRAPHIES

**SUNG PARK** is a Ph.D. student of Electrical Engineering at University of California Los Angeles. He received his MS from Carnegie Mellon University in 1997. His research interests are network simulation and modeling, low power embedded systems, and sensor network. His email and web addresses are <`spark@ee.ucla.edu`> and <`http://www.ee.ucla.edu/~spark`>.

**ANDREAS SAVVIDE** is a Ph.D. student of Electrical Engineering at University of California Los Angeles. He received his MS from University of Massachusetts, Amherst in 1997. His research interests are adhoc wireless network, embedded system, and sensor network. His email and web addresses are <`asavvide@ee.ucla.edu`> and <`http://www.ee.ucla.edu/~asavvides`>.

**MANI B. SRIVASTAVA** is an Associate Professor of Electrical Engineering at University of California Los Angeles. He received his Ph.D. from U. C. Berkeley in 1992. His research interests are low power VLSI circuits, embedded systems, and sensor networks. His email and web addresses are <`mbs@ee.ucla.edu`> and < `http://nesl.ee.ucla.edu/people/mbs/`>.