Using Sensor Networks for Highway and Traffic Applications

Tim Tau Hsieh

Sensor networks have been used for a variety of applications that include habitat/temperature monitoring, industrial sensing and battlefield awareness. However, many highway and traffic applications have not been tapped: primarily sensor networks for highway and traffic algorithms that alleviate generic problems such as highway congestion. This is due to the fact that sensor network technology is a very recent development. Since sensor networks are relatively new, not many applications have been explored in depth.

Utilizing the new generation of TinyOS micaboard mote sensors developed at the University of California-Berkeley, this article will focus on how to achieve the best possible data results from sensor network application and setup for traffic/highway goals. How to use Sensor-Network Graphs for optimal placement of sensors in a network so as to minimize work and to achieve the best possible, and most accurate, signal strength localization measurements will also be a primary focus. Also, discussed will be a method that optimizes the tradeoff between energy and accuracy using a variety of Activation Policies. Finally, simulations and distancing experiments of indoor and outdoor data are provided to encourage similar sensor work.

A brief history & initial transportation applications

Initially, the application and use of small-sized, low time-constant and high accuracy sensors were dominated by microwave detectors utilizing the Doppler Effect in the microwave range. According to Descamps et al, X and K band microwave Doppler sensors with printed antennas—using hybrid and Gallium Arsenide (GaAs) monolithic technologies—were developed for use strictly on cars and in guided transportation systems, mainly subways and railways. They were not only devoted to speed and distance measurements, but also to safety applications such as anti-locking braking systems, anti-skating systems and active suspensions.

The accuracy of these sensors actually depended on the nature of the ground. As a result, they were more or less accurate according to whether the ground was covered with snow, ice or water/rain. GaAs MEtal Semiconductor Field-Effect

Transistor (MESFET) technology was chosen primarily because it was conducive to high-performance and low-cost sensors for mass-production. Using the Doppler-shift principle, microwave technology senses moving objects by first sending a signal toward the roadway. When a vehicle passes through this pattern, some of the energy is reflected back to the unit at a different frequency (Clippard et al). This application of sensor networks can be used to detect mobile/moving targets such as vehicles and mopeds, especially approach-only and depart-only objects. The TinyOS sensor mote is based on a similar technology.

The TinyOS hardware

The primary sensor technology of interest here is TinyOS micaboard motes; miniaturized sensors that utilize TinyOS, an eventbased operating environment written in code similar to stylized C. They are compiled with NesC, a custom compiler often used with other embedded devices. The essential components of a small, 1.5”x 1.5”x 0.5”micaboard mote are: 1) the mote/sensor itself (that runs off a battery-supply), and 2) the sensorboard, a configurable sensor that allows communication and sensing between motes (shown in Fig. 1). Mica motes also use an ATmega103L micro-controller with a 4MHz CPU cycle frequency.

Wireless networking and communication between motes is done using a RFM TR1000 radio transceiver, which operates at the unique radio frequency of 916.50MHz. Mote communicate with one another by sending software packets through this transceiver/antennae.

In terms of software, TinyOS code is downloaded from a PC onto the mote’s 8kb flash memory (with 4KB of Static Random Access Memory (SRAM) as data memory) to run a variety of communication-based programs (sending packets,retrieving data, and turning on/off Light Emitting Diodes (LEDs)). As can be probably inferred, many TinyOS motes can easily be formed into a viable and reliable sensor network, with the transmission of packets carrying information (signal strength, location) being the transmitted data of interest.

With the mote’s RF wireless transceiver (having three LEDs for output), the analog-data interface and magnetometer located on the mica sensorboard can be used to detect magnetic materials, such as cars.

Magnetic uses

According to researcher Sinem Coleri, we know that we can only detect cars—magnetic material— with speeds in a specific interval; thus, solutions to this problem include adjusting the sampling rate high enough to detect the highest speed cars, and to consider the "absolute" magnetic field to detect lower speed cars.

Theories utilizing tree construction, node graphs and traversal algorithms optimize situations involving the micaboard mote "Base Station" in trying to detect cars in parking lots. Fortunately, for long-term monitoring applications such as these, power consumption for these mica sensors is not that much of a concern. In peak mode, the mote hardware consumes 19.5 mA, running about 30 hours on a battery. In inactive mode, the lifetime of the battery is nearly one year.

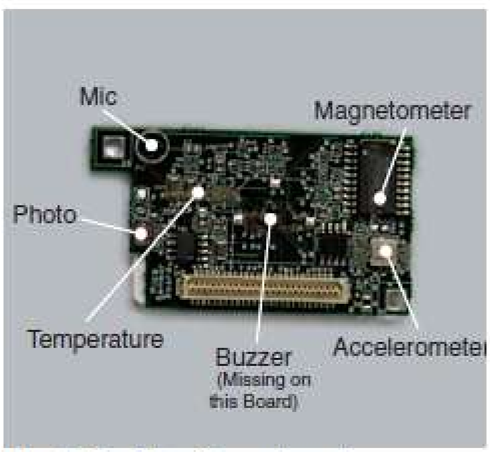

Fig. 1 The Mica Sensorboard

Fig. 2 The Mote with a radio antennae soldered on.

The primary goal of utilizing TinyOS sensor networks is to use signal strength readings to infer the distance between the motes. According to Whitehouse et al, the equation for signal strength is provided by:

$$y = (C\log(x) + v)$$

y being signal strength, *x* a distance vector, and *v*, some minor Gaussian noise. *C* is a mathematical parameter that can be optimized depending on a variety of constraints. In addition, a radio antenna can be attached to the micaboard mote (via soldering) so as to improve ranging and Radio Frequency (RF) communication (Fig. 2). The attachment of this radio helps greatly in calculating the signal strength through the formula just provided.

The utilization of this signal-strength/distancing data has a two-fold benefit in highway applications and analysis:

1. Surveillance metrics: The first usage is utilizing sensor detection in surveillance metrics. Varaiya and Coifman in their work using videotraffic detectors demonstrated its usefulness. The crux of their research essentially involves forming a vehicle reidentification algorithm for consecutive detector stations on a freeway, where “downstream” and “upstream” detector measurements were matched with a reproducible vehicle measurement, or vehicle signature. The city of Ann Arbor, Michigan followed a similar trend and has also utilized sensors (Microwave Sensors’ Model TC-20) to eliminate traffic problems such as gridlock in the busy streets of its Central Business District by monitoring upstream/downstream data.

2. Data for PeMS: A second use is monitoring traffic with sensors serving as information/data in the “front-end processors” (FEP). These processors retrieve data from freeway loops every 30 seconds in the Performance Measurement System (PeMS), a freeway performance measurement system for all of California, devised by the PATH research group led by Varaiya et al. Also, a wide variety of personnel (traffic engineers, managers, planners, travelers, researchers), depend on the realtime detector data provided by PeMS to form important operational decisions.

Sensor-network graphs

The optimal placement of sensors for getting the best possible and most accurate measurements is critically important to data transmission in sensor networks. The first important consideration is the minimization of work done. In order to minimize the amount of work/power we consume, we want to minimize the amount of data transferred in a network. A base station, or “sink” node, is a sensor node that takes as input detections from a variety of regions. It then generates output whenever a target, such as a vehicle, has been detected in any of the regions within a given time window. The problem, therefore, is essentially finding an optimal mapping of sensor nodes that minimizes the amount of data transferred among the regions to the base station.

Bonfils and Bonnet from the University of Copenhagen, Denmark propose a decentralized and adaptive solution to the sensor-placement problem.

Decentralized in the sense that each node should only maintain information about close-by or local nodes; ***Adaptive*** in the sense that operators and detections between nodes can be altered at any time, in an ad-hoc fashion. Their theory is centered around the notion of "cost," a function of the amount of data received and produced by a node. The cost is estimated from a set of nodes that receive data from an active node that transmits data. Of course, minimization of cost is the goal. They also define an oriented sensor network graph (SNG) as follows:

a) : a set of sensor nodes. In this set, p and q are elements of ζ.

b) : a set of communication links (edges) that connect the nodes in ζ.

c) *(p, q)* is a link between nodes *p* and *q*, an element of *λ*.

d) w_{pq}: a positive integer weight associated with the link *(p, q)* of *λ*.

The "cheapest" path between *p* and *q* (the path with the minimal cost) is denoted by $P_{min}(p, q)$. Where a path *P* = {*(p, x), (x, r), …, (y, s), (s, q)*} is between nodes *p* and *q*, and the cost is defined to be: where e _ P.

A graph traversal algorithm

The standard placement problem is a ***task assignment*** problem that is known to be NP-complete in the literature. The standard placement problem is also centralized and depends on information from "global" nodes. However, the solution proposed by Bonfils and Bonnet is decentralized and local. Their algorithm ***progressively refines the placement of operators (nodes) towards an optimal placement***. A node known as the "active node" is defined to be a node where a particular operation (data transmission) is executed. The procedure of the algorithm is as follows:

1) Evaluate the cost incurred by the execution of the operation at the active node.

2) Estimate the cost for the alternative assignments of the operation (the neighboring sensor nodes).

3) Compare the cost of the active node with that of the alternative neighboring nodes. The goal is to find a minimal cost from the setup.

4) Once the node with the lowest cost is found, transfer the operation to that node. This node becomes the new active node.

From this algorithm, therefore, we should be able to rig up a sensor network configuration on highways using TinyOS sensors that minimize the amount of data being transmitted. We, thus, minimize the power consumed.

However, sensor network topology is one of the main limitations to this algorithm. In the specific case of highways and traffic, if sensors were placed at different geographical heights, i.e. some on a mountain/building, while on the roadway level, data transfer would exhibit non-linear behavior. The ideal, and not very-realistic, case is to have data transferred along straight lines, all on the same topographical surface. Cost would then be simplified to a linear and directly proportional function of distance between nodes. Therefore, we desire the flattest sensor network topology available to simplify and optimize data transmission as much as possible. Fig. 5 shows Outdoor Percent and Absolute Error measurements taken from the TinyOS motes utilizing the aforementioned placement strategies. The flattest possible outdoor topography was used, and the Received Signal Strength Information (RSSI) MATLAB data confirmed the accuracy of this placement (Fig. 6).

The utilization of the graph-traversal theory also has significant application to detecting cars in a parking lot. In that particular case, the Base Station node serves as the “active node” and the data it wishes to transmit is a variety of packets that include ID and location. The nodes in the surrounding locations can use this ID data to determine the relative distances from one another and from the Base Station. The placement of the nodes so as to minimize data transmission can be done with the aforementioned algorithm. However, in an indoor parking-lot setting, the data may not be as accurate as data taken outdoors. Figures 3 and 4 show data taken from TinyOS motes placed indoors.

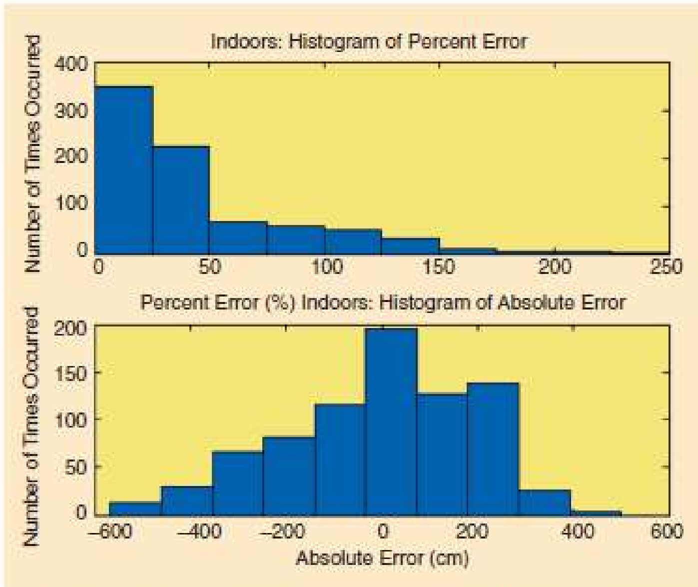

Fig. 3 Percent and Absolute Error Plots for indoor data

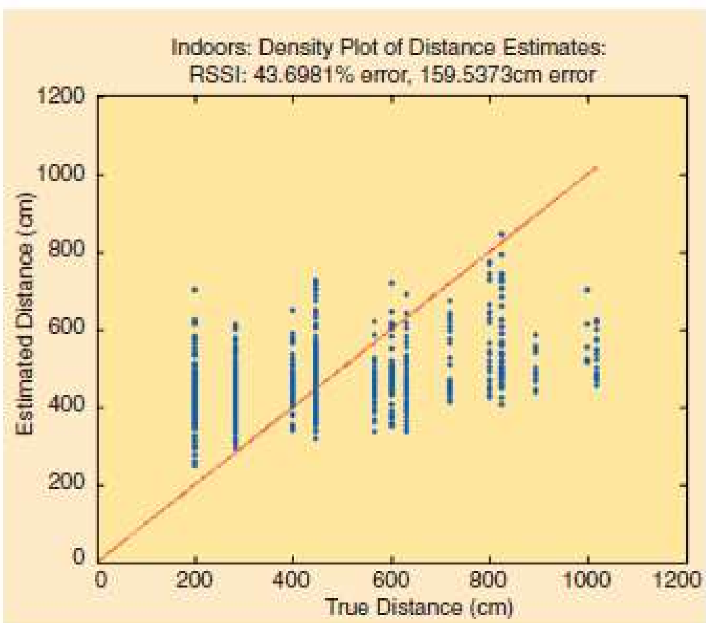

Fig. 4 RSSI Density Plot for indoor data

Activation policies for energy conservation

Design and engineering considerations in the placement of sensors are critical so as to maximize correct data and minimize error. For energy-efficient localization and tracking of mobile targets, such as cars using wireless sensor networks, gains in energy-savings come at the expense of increased accuracy in tracking, according to USC researchers Pattem, Poduri *et al*. Therefore, a direct tradeoff between energy/power consumption and the accuracy in which we can track objects or measure data can be immediately observed.

An intuitive way to save energy in nodes is to only turn on a subset of sensor nodes in a network, essentially, only the ones that are required. However, information provided by a small subset of nodes leads to an increased uncertainty in the sensed regions, i.e. there are less data points to confirm location and distance. Pattem, Poduri *et al* analyze these energy-quality tradeoffs by first proposing a quality metric and an energy metric, and then using those to develop four main tracking strategies. These strategies are utilized and simulated, and the results provide insight into the aforementioned tradeoffs:

1) *Naive Activation* (NA): All nodes in the network are in tracking mode all the time. Perhaps the worst energy efficiency, yet it serves as a useful baseline for comparison. Assuming *N* nodes, all *N* nodes are on, and the Power for the entire network is P = NS_ where *S* is the node sensing-range and a the sensed signal's decay exponent.

2) ***Randomized Activation* (RA):** Each node is on with a probability p. A fraction of Nodes, *p*N, will be on, and the network's Power is $P = pNS^a$.

3) ***Selective Activation based on prediction* (SA)**: Only a small subset of nodes is in tracking mode at any given point in time. They are intelligent in that they also predict the "next" position of the mobile target (e.g. car) and hand over the tracking to nodes that are best placed to track the target in this "next" position. If we define Xp to be the predicted target position, then the sensor nodes within a radius Sp around Xp are in tracking mode at any given point in time. If ρ is the density of sensor deployment, then π(Sp)2ρ is the number of nodes that are on, and the collective Power is P = π(Sp)2ρSa.

4) ***Duty-cycled Activation* (DA):** The entire sensor-network periodically turns off and on with a regular duty cycle. One interesting feature of DA is that it can be used in conjunction with any of the other activation strategies (SA, NA, RA). T_D is the period of the cycle, t_{ON} is the on-time, and n_sU be the average number of tracking sensors in the accompanying activation strategy (SA, NA or RA) U. Then the number of modes that are on is = $(nsUt_{ON}) / T_D$ and the collective Power is then

$P = (nsUt_{ON}S^{a}) / T_D$

The variety of simulations Pattem, *et al* run include simulation of a virtual large scale sensor network on a 200 x 200 unit area with a random placement of sensors and a density of sensor deployment to be ρ = 1 sensor/unit area (a total of 40,000 nodes). For naïve activation, the tracking error decreases as the sensing range *S* increases. The same happens with random activation, at varying values of *p*. If *p* is decreased for random activation, then the tracking quality is also significantly decreased. For selective activation, the tracking error was quite high for Sp = S. Selective activation with a Sp = 1.5S performs nearly as well as naive activation.

When naive activation, random activation and selective activation are all compared together, the dominating and best strategy appears to be selective activation with a fairly high Sp. It essentially is the best in terms of the best tradeoff between low-error as well as low energy/power expenditure. Selective activation was also shown to have four orders of magnitude savings in energy compared to naive activation or random activation, with optimal settings. However, a feasible value of Sp must be chosen, and it depends on the mobility of the target observed.

With duty-cycled activation, the best out of the three (selective activation, random activation or naive activation) must be used to obtain optimal results. In this case, if selective activation is used in conjunction with duty-cycled activation, then we have the best possible tracking strategy. The combined strategy is, of course, duty-cycled selective activation. It can be appropriately adjusted with alternating values of t_{ON} or T_D. Those two variables essentially serve as “tuning knobs” of sorts.

From the previous analysis of tracking strategies, it can be inferred that the optimal strategy for highway and traffic applications is also perhaps selective activation. It would be best if all the TinyOS micaboard sensors could be equipped with intelligent packet-routing capabilities that would allow them to communicate the “next” area/location of where a car is headed. Once the car heads to that “next” location, the micaboard motes in the area “left” by the car would have a mechanism to turn themselves off, while the motes in the “next” area would be able to turn themselves on. This would propagate successively as a sequence of circles with radius Sp to achieve optimal measurement accuracy and minimal tracking error.

Also, power and energy consumption would be in turn minimized. For this setup of successive circles, it is best perhaps to utilize an omni-directional ultrasound micaboard, in order to sense the mobile target in all directions. If we use this along with a duty-cycled activation, and adjust the “tuning knobs” of t_{ON} or T_D accordingly, then we can be sure we are obtaining the best possible tradeoff between tracking error and energy/power expenditure.

In summary

Sensor networks have a wide variety of applications, from monitoring environmental data, to observing natural phenomena, and from various target tracking to even prevention of terrorist attacks according to Tubaishat *et al*. By embracing recent sensor network technology, many practical applications, both highway and non-highway related, can be discovered. Further work and research for highways is encouraged. Other potential developments include prevention of car collisions, pedestrian safety and lane-maintenance. All are very interesting sensor-network research topics that will improve the safety and efficiency of our highways for the future.

Read more about it

• Varaiya, Pravin., Coifman, Benjamin.: Improving Operations Using Advanced Surveillance Metrics and Existing Traffic Detectors. California PATH Research Report. UCB-ITS-PRR-2002-2. January 2002.

• Descamps, P., Bouazza, F., Sadek, S., Vindevoghel, V.: Advanced Microwave Sensors for Transportation Applications. Transportation Systems: theory & application of advanced technology, Vol.1 Oxford : Pergamon, 1995. 25-30.

• Clippard, Denise.: Overhead Microwave Detectors Cut The Loop. Traffic Technology International, 1996. 118-120.

• Whitehouse, Kamin., Klemmer, Scott., Waterson, Sarah.: Towards a Location-Based Context-Aware Sensor Infrastructure. UC-Berkeley CS Department, GUIR. 2001. http://guir.berkeley.edu/location.

• Varaiya, Pravin.: Freeway Performance Measurement System, PeMS v3, Phase 1: Final Report. California PATH Research Report. UCBITS-PWP-2001-17. October, 2001.

• Bonfils, Boris Jan., Bonnet, Philippe.: Adaptive and Decentralized Operator Placement for In-Network Query Processing. Department of CS, University of Copenhagen, Denmark. Proceedings of 2nd International Workshop on Information Processing in Sensor Networks (IPSN 2003). Lecture Notes in Computer Science, Springer. April 2003.

• Coleri, Sinem.: TinyOS and Sensor Network Applications. Power-Point Presentation. http://wow.eecs.berkeley.edu. January 2002.

• Pattem, Sundeep., Poduri, Sameera., Krishnamachari, Bhaskar.: Energy-Quality Tradeoffs for Target Tracking in Wireless Sensor Networks. Dept of EE and CS at University of Southern California. ISPN 2003. Lecture Notes in Computer Science, Springer, April 2003.

• Tubaishat, Malik., Madria, Sanjay.: "Sensor Networks: An Overview." IEEE Potentials Magazine. April/May 2003. 20-23.

• Whitehouse. Kamin. My Research Files. http://www.cs.berkeley.edu/~kamin. April 2000. 12. Johnson, Colin, R.: Companies Test Prototype Sensor Nets. EE Times online. January 29, 2003. http://www.eetimes.com

About the author

Tim Tau Hsieh is currently an Electrical Engineering and Computer Science student at The University of California, Berkeley. He has done research with TinyOS, Sensor Networks, and applying Sensor Networks for use on Highways for PATH - Partners for Advanced Transit (http://www.path. berkeley.edu). He has worked at Programmable Silicon Solutions, a small RFIC and Flash Memory Design House, and will start a job at Lockheed Martin next Fall.