SVATS: A Sensor-network-based Vehicle Anti-Theft System

Hui Song
Department of Computer Science
Frostburg State University
Frostburg, MD 21532
Email: hsong@frostburg.edu

Sencun Zhu and Guohong Cao
Department of Computer Science & Engineering
The Pennsylvania State University
University Park, PA 16802
Email: {szhu, gcao}@cse.psu.edu

***Abstract*—Today vehicle theft rate is very high, thus tracking/alarming systems are being deployed with an increasingly popularity. These systems however bear some limitations such as high cost, high false-alarm rate, and easy to be disabled. This paper describes the design, implementation and evaluation of a Sensor-network-based Vehicle Anti-Theft System (SVATS) to address these limitations. In this system, the sensors in the vehicles that are parked within the same parking area first form a sensor network, then monitor and identify possible vehicle thefts by detecting unauthorized vehicle movement. When an unauthorized movement is detected, an alert will be reported to a base station in the parking area, which sends warning messages to the security office. This paper focuses on the technical issues specific to the system such as topology management, theft detection, and intra-vehicle networking.**

I. Introduction

Today, vehicles have been an essential part of our daily life. Unfortunately, we are also facing the high possibility of vehicle theft. For example, based on an article [1] published in USAToday, the National Insurance Crime Bureau reported that nearly 1.3 million vehicles were stolen in 2003. The vehicle theft rate held steady at about 433 cars stolen per 100,000 people in 2003.

Because of the high theft rate, vehicle tracking/alarming systems become more and more popular. Generally, these systems can be classified into three types: *lock devices*, *alarm systems*, and *vehicle tracking/recovery systems*. The commonly used lock device is the steering wheel lock. Although it is relatively cheap, it is inconvenient to use and may be easily disarmed by skilled thieves. Car alarm systems (prices range $100 to $500) are very popular these days. However, the vast majority of blaring sirens are false alarms and people have been used to the alarms and do not care about them.

The commonly used vehicle tracking/recovery systems are based on radio signals such as the Lojack tracking system, the ProScout GPS Vehicle Tracking System, the TravelEyes2 Vehicle Tracking System and so on. After a vehicle has been stolen, the owner can report the problem to the police or the GPS tracking office. The wireless transmitter or the GPS device in the car will send wireless signals which can be picked up by the tracking device. The wireless signals can be used to pinpoint the location and lead police to rapid recovery. However, there are several disadvantages. First, these systems have high cost of $500 to $1300. Although the up-front purchase price keeps decreasing, the maintenance cost remains high. For example, these systems often come with a monthly monitoring fee. Second, GPS-based systems do not work indoors and terrain interference may occur in dense urban areas. Third, GPS-based tracking systems are easy to defeat since the thief knows where device is located. The thief can simply break off the antenna or cover it with metal, and then the GPS tracking system will become useless.

To address the limitations of existing vehicle tracking/alarming systems, we propose a Sensor-network-based Vehicle Anti-Theft System (SVATS). In SVATS, each vehicle is equipped with some sensors. All sensors in vehicles parked in the same parking area form a sensor network. For each parking area, one separate sensor network is formed and one base station (BS) is installed. SVATS relies on the sensors to detect vehicle theft and notifies the police through the BS.

The rest of the paper is organized as follows. In Second II, we give the system overview of SVATS. Section III presents the design of SVATS and techniques used in SVATS. Section IV concludes the paper.

II. System Overview

In SVATS, each vehicle has a wireless sensor node which can be connected to the power source of the vehicle. All sensors in vehicles parked in the same parking area such as shopping centers, schools, hospitals, airports, residential areas, form a sensor network. For each parking area, one separate sensor network is formed and one base station (BS) is installed.

Within a sensor network, each node is monitored by its neighbors, which identify possible vehicle thefts by detecting unauthorized vehicle movement. For example, suppose Emily comes back home and wants to park her car in the residential parking lot. Before she leaves the car, she powers on the sensor node in her car by a remote controller. The sensor node broadcasts an authenticated "join" message to sensors in the neighboring cars. After joining the network, it periodically broadcasts an authenticated "alive" message to its neighbors. Commanded by the remote controller, it sends an authenticated "leave" message to neighbors when the car leaves; the sensor is then turned off. If a thief moves the car, without sending an authenticated "leave" message, the neighboring sensors can detect the car movement. If the thief destroys the sensor in the car, the neighbors will not receive authenticated "alive" messages from the sensor, thus detecting the abnormal phenomenon. They will report the problem to the BS, which in turn automatically sends a warning message to the security officer. The vehicle owner can also be notified at their choice.

The aforementioned basic SVATS system is enough to detect stolen vehicle. To track the stolen vehicle, we enhance SVATS by using the wireless nodes or access points deployed along major streets and around the intersections. Note that this roadside wireless access points may not be an extra requirement of SVATS. Many vehicular ad hoc networks [2], [3], [4] need to access this road side devices to improve road safety and support many commercial applications [5]. This roadside wireless access points can be used to communicate with the sensors within the passing-by vehicles. In case a car is stolen, the sensor node within the car can detect its own unauthorized movement by using movement sensors or by measuring neighboring car's sensor signal, and hence report problems to the roadside wireless devices. In this way, the vehicle can be tracked city-wide as long as it is within the area where SVATS system has been deployed.

Since the sensor is attached to the vehicle power, its position is known and may be destroyed by the thief, and then cannot report problems for tracking. To address this problem, we deploy more sensor nodes, referred to as the *slave sensors*, inside each vehicle. The slave sensors should be put at several hidden places inside the vehicle so that the thief cannot locate them in a short time. Slave sensors are used to monitor the original sensor node, referred to as the *master sensor*, and to report vehicle theft when master sensor is destroyed.

III. SYSTEM DESIGN

SVATS includes the following four components: network topology management, vehicle theft detection, intra-vehicle networking, and alert reporting. In this paper, we only discuss techniques for the first three components due to space limit.

A. Network Topology Management

Vehicles join/leave the parking lot frequently, and hence the network topology keeps changing. We rely on power control techniques to maintain a network topology so that a vehicle has enough neighbors to monitor it.

We formulate the topology management as a localized, distributed process. Each sensor checks its neighbor number periodically (and adjusts its transmission power level if necessary) to maintain a desired number of neighbors with the minimum needed transmission power level to reduce interference to other nodes. Specifically, it has three phases: initial power-level estimation, neighbor discovery and neighbor maintenance. The first two phases are executed when a node first joins the network; whereas the last phase is executed periodically after the sensor has joined the network.

Initial Power-level Estimation After joining the network, the sensor first estimates its transmission power level. Although using the estimated transmission power level cannot guarantee that the sensor will find enough number of neighbors, this phase can speed up the neighbor discovery process.

After a car has been parked, the sensor node inside the vehicle (e.g., node A) is triggered to power on, and it listens to and collects "alive" messages from neighboring nodes. After A has collected enough "alive" messages, it counts the number of neighbors that it can hear. In addition, A retrieves the transmission power levels from the "alive" messages and order them, from low to high, to form a power-level list. If the number of neighbors that it can hear is larger than the desired number of neighbors, the sensor can choose a power level that meets the desired number of neighbors. Otherwise, the sensor should at least use the maximum power level of all received "alive" messages as the estimated transmission power level.

Neighbor Discovery With the estimated initial transmission power level, the joining node initiates the neighbor discovery phase by broadcasting a *join* packet to its neighbors. The *join* packet includes a neighbor list; i.e., the list of nodes that it can hear in the previous phase. When a node broadcasts a *join*, some neighbors that are not in the neighbor list can still receive the *join* packet, which are referred to as *unidirectional neighbors*. The nodes that receive *join* will check whether they are within the neighbor list. If not, they just ignore the join request. Otherwise, they mark themselves as a neighbor node of the new joining node and send "reply".

If the joining sensor node can receive enough replies, the neighbor discovery process terminates. Otherwise, it has to find more neighbors. In this case, it has to increase its transmission power level and send a neighbor discovery message. This process is repeated until it either finds enough number of neighbors or reaches the maximum transmission power level. It may take a long time to find enough neighbors if the transmission power level is linearly increased. Using the MICA2 mote [6] as an example, which has 255 power levels, the joining node may have to try 255 times to find enough neighbors in the worst case.

To reduce the neighbor discovery time, the joining node increases the transmission range using a strategy similar to binary-search, hoping to pinpoint the minimum needed power level quickly. Unfortunately, it is difficult to adjust the transmission range in a binary-search way for the current generation sensor nodes such as MICA2 motes. The reason is that the radio transmission power level (or signal strength RSSI) is measured by ADC counts (from 1 to 255) in stead of by dBm. Further, the relationship between transmission power level and transmission range is nonlinear.

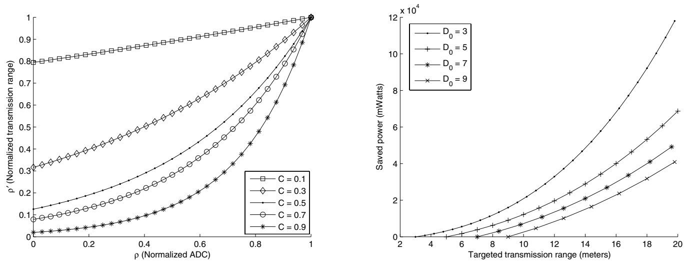
Next, we derive a formula to facilitate adjusting the transmission range in a binary-search way based on the *log normal shadowing radio model* [7]. This model is a statistical model for variations in the received signal amplitude due to blockage. It combines the effect of both path loss and shadowing.

Let the maximum transmission range be D and the corresponding ADC counts be ADC_{max}. Now suppose we need to jump to the transmission range of $\rho \cdot D$ in a binary-search (ρ can be $\frac{1}{2}$, $\frac{1}{4}$, $\frac{3}{4}$, $\frac{1}{8}$, $\frac{3}{8}$, and so on), we can adjust the transmission power level in terms of ADC counts accordingly following Eqn. 1.

$$ADC_\rho = ADC_{max} + 10\beta\log\rho \cdot M. \qquad (1)$$

In Eqn. 1, β is called the path loss exponent, which is often between 2.7 to 5. M is a constant. Both β and M are usually empirically determined by field measurement.

Preliminary Results: Our preliminary analytical results on binary-search in the ADC counts (*Method 1*), binary-search in the transmission range (*Method 2*), and linear-search in the transmission range (*Method 3*) are shown in Fig. 1. From this figure, we can observe that (1) using Method 1, the transmission range changes nonlinearly; (2) Method 2 is much more energy-efficient than Method 3. Due to space limits, we put the details on how we derived Eqn. 1 and the details of preliminary analysis in a technical report [8].



(a) (b)

Fig. 1. Mathematical Analysis results. Subgraph (a) shows that Method 1 (x axis) results in nonlinear changes in terms of transmission range (y axis). Subgraph (b) shows the amount of saved power using Method 2 compared to Method 3. (D_0 is the estimated transmission range; and C is a constant that can be collected by experiments.)

If the joining node still cannot find enough neighbors after it increases its power level to maximum, it asks those unidirectional neighbors to join. In this round of join request, if a unidirectional neighbor is willing to serve as a monitoring node, it can notify the joining node. Since the unidirectional neighbor cannot send data directly to the joining node, it either increase its power level or ask other nodes to forward its reply.

Neighbor Maintenance To maintain a desired number of neighboring nodes, each sensor checks its number of neighbors periodically and adjusts its transmission power level if necessary.

B. Vehicle Theft Detection

In this section, we present two vehicle theft detection techniques: count-based and statistical-based.

1) Count-based Theft Detection: After joining the sensor network, the sensor node keeps broadcasting "alive" message (also referred to as node advertisement) periodically. Each neighbor can measure the signal strength (or RSSI) of the received alive message. The RSSI is associated with the sending node's transmission power level and the distance between the sender and the receiver. If a monitoring node does not receive a certain number of node advertisement messages from the neighbor that it is monitoring, the monitoring node would assume that the neighbor has moved. The theft detection consists of three components: theft detection, theft attestation, and distributed voting.

Theft detection: For each monitee, a monitor keeps a counter that counts the number of node advertisements (NA) missed by that node. The count (NA) is increased by one when the advertisement timer runs out. When NA reaches a MAX_ADV_MISSES, the monitor suspects that the monitee as un-reachable and initiates the verification process.

Theft attestation: The attestation phase is necessary because situations other than theft could also result in the miss of node advertisement from the given monitee. For example, passing-by objects (e.g., vehicles or human beings) that temporarily block the communication path between the monitor and the monitee.

Using this component, the monitor can attest the vehicle theft and reduce the false alarm rate. The monitor either confirms or voids its detection based on the attestation results. More specifically, the monitor sends a challenge to the monitee and waits for a response. If the monitor does not receive a response from the monitee within a given timeout period, the monitor claims that the monitee is moved and a vehicle theft is detected. The challenge-response process can be executed several times. If none of the challenges gets through, the attestation confirms a vehicle theft detection; otherwise, the theft detection should be canceled.

Distributed voting: The attestation at a single monitor node could still result in a false alarm due to the unreliability of the measurement method. In this phase, every node that confirms a detection should broadcast a detection announcement to others. Based on the received distinct detection announcements, each monitor makes a final decision on whether the monitee is stolen or not. In our system setting, if a monitor receives three or more detection announcements from different nodes, it claims that a vehicle theft detection is verified. In case that there are less than three monitors for a monitee, a monitor can still confirm a detection if it detects the theft continuously for several rounds (say three) of testing. After the detection has been verified, it sends an alert message to the BS. Note that the voting is distributed; i.e., the verification is done at each node independently. The distributed property avoids the security weakness of depending on a single node to make the final decision.

2) Statistical-based Theft Detection: The count-based scheme has one disadvantage: The detection time is relatively long. The monitors start to miss node advertisements sent from the monitee only when the vehicle has moved out of the transmission range. Thus, the vehicle theft will not be detected until the vehicle has been moved for a distance of the sensor's transmission range, which could be as large as 70 meters for Mica2 motes. As a result, the response time for theft detection is long.

To reduce the movement (or theft) detection latency, we propose to use a statistical method called *unpaired observations* [9]. This technique has been widely used in testing a hypothesis based on the difference between sample means. If the sensor node moves to a different location, the distributions of the RSSI values measured by a same neighbor (from the same location) will be different. Thus, if a monitoring node makes *before* and *after* measurements on the vehicle that it is monitoring, it will be able to tell the difference between *before* and *after* measurement pairs (of RSSI values) and determine whether that vehicle has moved or not.

Unpaired observation involves two phases: training and detection. In the training phase, each sensor node aggregates the RSSIs of the neighbor that it is monitoring and computes the mean and the standard deviation of the values. This phase could be executed and finished within a short period of time right after the sensor first joins the network. In the detection phase, each sensor node measures the RSSI data periodically received from the neighbor that it is monitoring and uses unpaired observations to determine whether the RSSIs have significant changes. Specifically, if the confidence interval of the RSSI data includes zero, the difference is not significant at a certain confidence level. Otherwise, the difference is significant and the sensor node concludes that the vehicle it is monitoring has moved.

A monitoring node can make a decision on whether another vehicle has moved or not on its own. However, the detecting node may make a wrong decision due to measurement errors or malicious attacks. To address these problems, we can use the generalized extreme studentized deviate (GESD) algorithm to preprocess the data before applying the unpaired observations scheme. The purpose is to detect and filter out the outliers introduced by measurement errors or attacks. Another issue is that the join/leave of a vehicle between the two sensors may affect their measurements. To reduce false alarms, these two sensors should restart over the training phase after a join/leave event.

To further decrease the false-positive rate, distributed voting techniques can be used. To do this, every node that confirms a detection should broadcast a detection announcement to others. Based on the received distinct detection announcements, each of them makes a final decision on whether the accused one is stolen or not. After the detection has been verified, it sends an alert message to the BS. Note that the voting is distributed; i.e., the verification is done at each node independently. The distributed property avoids the security weakness of depending on a single node to make the final decision.

Distance Measurement Normality Test: The unpaired observations technique is critical for effective theft detection. This technique, however, relies on the assumption that the observations come from a population that has normal distribution. To verify the normality of the distance measurements, we conducted an experiment using Mica2 motes. In the experiments, two sensors are placed 5 meters away from each other. One sensor broadcasts beacon signal periodically and the other node measures the received beacon RSSI. Fig. 2(a) shows the histogram of the measured RSSI, which indeed follows a normal distribution.

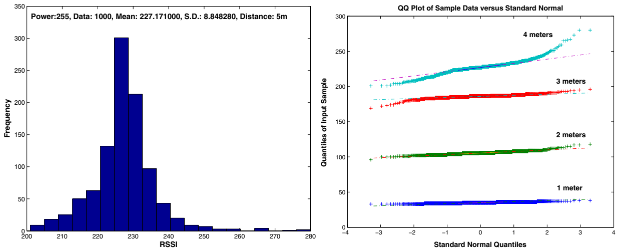
Fig. 2(b) illustrates the quantile-quantile (Q-Q) plot of RSSIs at different distances. Q-Q plot displays the sample quantile of RSSIs versus the theoretical quantile from a normal distribution. If the distribution of RSSIs is normal, the plot should be close to linear. From Fig. 2(b) we can see that the RSSIs for different distances are close to linear. Based on Fig. 2, we claim that the distribution of distance measurements follows a normal distribution.

Experimental results: Fig. 3 shows the experiment setup where vehicle 4 is monitored by five other sensors. We drive away vehicle 4 without sending a leave message. As illustrated in Fig. 4, the detection delay of the statistical-based scheme is much shorter than that of the the count-based scheme. Due to space limit, more results are shown in [8].

C. Intra-Vehicle Networking

Normally, master sensors can be used to communicate with roadside wireless access points for tracking the stolen vehicle. Since the sensor is attached to the vehicle power, its position is known and may be destroyed by the thief. At this time, slave sensors can be used to track the stolen vehicle. The slave nodes are normally in sleep. After the master sensor node sends “join” message, the slave nodes start to monitor the master sensor. After the master sensor sends “leave” message, the slave nodes will go to sleep again. If the thief destroys the master sensor before it sends “leave” message, these slave nodes will report problems to roadside wireless points that the vehicle is stolen. Since the slave sensors have to be hidden in some places hard to find, they have to run their own power. As a result, energy efficiency of these nodes is an important issue.

To extend the battery life, the slave sensors should keep sleep most of time. They also need to monitor the status of the master sensor in case the master sensor is destroyed. There are many existing work on designing sleep schedules to save power. However, SVATS is different from a typical sensor network, where sensors are used to regularly collect



(a) The histogram (b) The Q-Q plot

Fig. 2. Test the normality of the distance measurements. One figure shows the histogram of the measured RSSI and the other one shows the quantile-quantile plot of the RSSIs at difference distances.

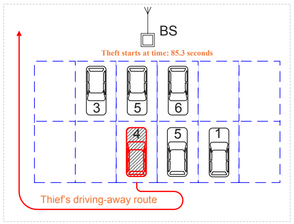

BS
Theft starts at time: 85.3 seconds
3
5
6
4
5
1
Thief's driving-away route

Fig. 3. Experiment setup

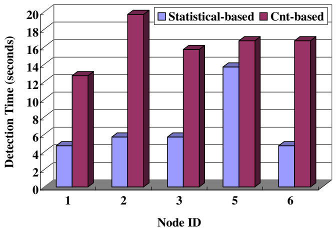


Fig. 4. Detection delay

information from the deployed field. Instead of monitoring the external environment, the slaves monitor the beacon messages sent from the master sensor, and hence the slaves must be awaken when the master sensor is sending messages such as alive, leave, join, etc. As a result, the sleep scheduling of the slave sensors should be synchronized to the master sensor's message sending behavior.

In our solution, each time the master sensor turns on, it wakes up the slaves. Gu and Stankovic [10] proposes a remote radio triggered hardware, which extracts energy from specific radio signals without using an internal energy source, to provide wake up signals. This technique can be used here to wake up the slaves. Other remote wake up techniques can be used here as well.

After being waken up, all slaves synchronize their clocks to that of the master sensor, using time synchronization protocol such as RBS [11]. Later, the slaves can periodically synchronize to the master, since they will hear the beacon message at each sleep-wakeup cycle. Thus, we can assume that the slaves and the master have approximately synchronized clocks.

Then the master sensor announces its beacon interval I_b and the expected cycle interval N (in terms of number of beacon intervals). Based on I_b and N, each slave calculates its wakeup time utilizing a random number generator (RNG). More specifically, the slave first generates a random number based on its node id, denoted as RNG(id); then its wakeup time within the cycle is calculated as: $(\text{RNG}(id) \bmod N) * I_b$.

The wakeup times assigned to all the slaves are not optimal due to the limitations of RNG. Some slaves may end up with the same wakeup time. To mitigate this problem, we should pick a good RNG and choose a prime number for N. Another reason for the sub-optimal sleep scheduling is due to a unique characteristic of slave sensors: They only passively listen and do not exchange packets with others. One reason of not exchanging packets with each other is to protect these slave sensors. Since they only passively listen during normal time, the thief cannot locate them based on electrical signal.

Since many slaves may be at sleep when the master sends the leave message, they will not be able to receive the leave message successfully. To solve this problem, the master sensor is required to send the leave message at the beacon interval (as sending the alive message) for a period of time (e.g., for two sleep-and-wake cycles). In this way, every slave will receive the leave message at least once. Note that, when a slave sensor receives a leave message, it will power itself off to save power.

IV. Conclusions and Future Work

This paper presented a sensor-network-based vehicle anti-theft system (SVATS), which can detect unauthorized vehicle movement and track the stolen vehicle. SVATS is a large system; clearly, its success will require more than techniques. In this paper, we focused on the technical issues specific to the system such as connectivity management, movement detection, and intra-vehicle networking. SVATS can be deployed incrementally. Its nice features such as rapid response and resilience to attacks will motivate many people to install SVATS sensor nodes. If successfully deployed, SVATS will become the largest practical sensor network application, and will provide a platform for testing many of the techniques proposed for wireless sensor networks.

Current SVATS still has several limitations. The biggest one is caused by network partitions in a sparse parking lot. In the extreme case no neighbors can be found even if a sensor has tried its maximum power level. This is not a concern if the sensor can communicate with the BS directly. For a large parking space, this may not always hold. As the result, this vehicle cannot receive any protection.

One possible solution is to deploy extra sensors inside the parking area securely (or invisibly), which can be used to monitor the parked vehicles and forward alert messages. However, the cost of the system will be increased. Note that even when there are not enough neighbors, the slave sensors inside the stolen vehicle can still provide tracking service as long as there are existing road side wireless nodes. Also, the slave sensors can send the alert to sensors in passing by vehicles, which can carry the alert to a BS or directly send to the police through its vehicular network.

The security of SVATS is critical for its success because the goal of an attacker will be to evade the detection of the system. There may also be privacy concerns, and will be our future work.

Acknowledgment

This work was supported in part by the National Science Foundation (CNS-0524156, CNS-0519460), and Army Research Office (W911NF-05-1-0270 and W911NF-07-1-0318).

References

[1] USA Today. (2004) Top car-theft areas in each state. [Online]. Available: http://www.usatoday.com/news/nation/2004/11-29-car-thief-table.htm

[2] V. Bychkovsky, B. Hull, A. Miu, H. Balakrishnan, and S. Madden, "A measurement study of vehicular internet access using in situ wi-fi networks," *ACM Mobicom*, 2006.

[3] B. Hull, V. Bychkovsky, Y. Zhang, K. Chen, M. Goraczko, A. Miu, E. Shih, H. Balakrishnan, and S. Madden, "Cartel: A distributed mobile sensor computing system," *ACM Sensys*, 2006.

[4] D. Jiang, V. Taliwal, A. Meier, and W. Holfelder, "Design of 5.9 GHz DSRC-Based Vehicular Safety Communication," *IEEE Wireless Communications Magazine*, October 2006.

[5] J. Zhao and G. Cao, "VADD: Vehicle-Assisted Data Delivery in Vehicular Ad Hoc Networks," *IEEE INFOCOM*, April 2006.

[6] Crossbow Technology Inc., "Wireless sensor networks," in *http://www.xbow.com/*, Accessed in November, 2004.

[7] S. Y. Seidel and T. S. Rapport, "914 mhz path loss prediction model for indoor wireless communications in multi-floored buildings," *IEEE Trans. on Antennas & Propagation*, Feb. 1992.

[8] H. Song, S. Zhu, and G. Cao, "Svats: A sensor-network-based vehicle anti-theft system," Networking and Security Research Center, Department of Computer Science ang Engineering, Pennsylvania State University, Technical Report NAS-TR-0076-2007, August 2007.

[9] NIST/SEMATECH, "e-handbook of statistical methods," http://www.itl.nist.gov/div898/handbook/toolaids/pff/prc.pdf.

[10] L. Gu and J. A. Stankovic, "Radio-triggered wake-up for wireless sensor networks," *Real-Time Syst.*, vol. 29, pp. 157–182, 2005.

[11] J. Elson, L. Girod, and D. Estrin, "Fine-grained network time synchronization using reference broadcasts," *SIGOPS Oper. Syst. Rev.*, 2002.