

# Achieving Long-Term Surveillance in VigilNet

Tian He<sup>†</sup>, Pascal Vicaire, Ting Yan, Qing Cao<sup>‡</sup>, Gang Zhou, Lin Gu,  
Liqian Luo<sup>‡</sup>, Radu Stoleru, John A. Stankovic, Tarek F. Abdelzaher<sup>‡</sup>

Department of Computer Science  
University of Virginia, Charlottesville 22903

**Abstract**—Energy efficiency is a fundamental issue for outdoor sensor network systems. This paper presents the design and implementation of multi-dimensional power management strategies in VigilNet, a major recent effort to support long-term surveillance using power-constrained sensor devices. We integrate a novel tripwire service with an effective sentry and duty cycle scheduling in order to increase the system lifetime, collaboratively. Through extensive system implementation, we demonstrate the feasibility to achieve high surveillance performance and energy efficiency, simultaneously. We invest a fair amount of effort to evaluate our architecture with a network of 200 XSM motes in an outdoor environment, an extensive simulation with 10,000 nodes, as well as an analytical probabilistic model. These evaluations demonstrate the effectiveness of our integrated approach and identify many interesting lessons and guidelines, useful for the future development of energy-efficient sensor systems.

## I. INTRODUCTION

VigilNet is a recent major effort to support long-term military surveillance, using large-scale micro-sensor networks. Besides requirements of accurate target tracking and classification [1], one of the key design goals of VigilNet is to achieve long-term surveillance in a realistic mission deployment. Due to the small form factor and low-cost requirements, sensor devices such as the XSM motes [2] are normally equipped with limited power sources (e.g., two AA batteries). Moreover, because of the hostile environment and a large number of nodes deployed, currently it is not operationally and economically feasible to replace the power source without introducing enormous effort and elements of risk to the military personnel. In addition, the static nature of the nodes in the field prevents the scavenging of the power from ambient motion or vibration [3][4]. The small form factor and possible lack of the line of sight (e.g., deployment in the forest) make it difficult to harvest solar power. On the other hand, a 3~6 month system life span is essential to guarantee the effectiveness of normal military operations, which necessitates a 12~24 fold extension of the normal lifetime of active sensor nodes. Consequently, it is critical to investigate practical approaches of spending the power budget effectively.

Many solutions have been proposed for energy efficiency at various levels of the system architecture, ranging from the hardware design [5][2], coverage [6][7][8][9] MAC [10][11][12], routing [13][14][15], data dissemination [16], data gathering [17][18], data aggregation [19][20],

data caching [21], topology management [22], clustering [23], placement [24] [25] to energy-aware applications [26][27]. Instead of focusing on a single protocol, our answer to energy efficiency is an integrated multi-dimensional power management system. Our contributions, presented in this paper, are identified in the following aspects: 1) Our design is validated through an extensive system implementation: VigilNet – a large-scale sensor network system delivered to military agencies. 2) VigilNet takes a systematic approach, and the energy efficiency is not narrowly accounted for within a single protocol. We propose a novel tripwire service, integrated with an effective sentry and duty cycle scheduling to increase the system lifetime, collaboratively. 3) Tradeoffs are investigated to meet requirements of both surveillance performance and the network lifetime. We present a complete system with 40,000 lines of code, running on motes, that achieves performance and energy efficiency simultaneously. 4) We devote considerable effort to evaluate the system with 200 XSM motes in an outdoor environment and an extensive simulation of 10,000 nodes, in order to identify a set of useful lessons and guidelines for future research.

The remainder of the paper is organized as follows: Section II categorizes power management features for different application scenarios. Section III describes the power management requirements in VigilNet. Section IV introduces three power management strategies utilized in VigilNet namely, sentry service, tripwire service and duty cycle scheduling. Section V describes the integrated power management architecture in VigilNet. Section VI briefly discusses some additional energy efficient techniques applied in VigilNet. Section VII addresses the tradeoff between energy efficiency and network performance. Section VIII details the system implementation. Section IX provides the evaluation of a network of 200 XSM motes as well as an extensive hybrid simulation with 10,000 nodes. Finally, Section X concludes the paper.

## II. BACKGROUND

Power management is by no means a stand-alone research issue. It can be dramatically affected by the underlying system configuration and by the application requirements. These include the form factor [28], hardware capability [5], possibility of energy scavenging [4][29], network/sensing topology and density [6], link quality [30], event patterns, node mobility, availability and accuracy of time synchronization [31], real-time requirements and the nature of the applications [26]. At the hardware level, multi-level sleep modes in the low power

<sup>†</sup>Tian He is now with University of Minnesota tianhe@cs.umn.edu

<sup>‡</sup>Qing Cao, Liqian Luo and Tarek Abdelzaher are now with University of Illinois, Urbana-Champaign

microcontroller [5] enable software to control the rate of power dissipation. Fine-grained power control [2] allows applications to activate hardware modules incrementally. Radio wakeup circuits [32] achieve passive vigilance with a minimal power draw. Energy scavenging [3] is also possible for some application scenarios, where ambient energy can be harvested. Sensing coverage schemes [6][7] exploit redundancy in the node deployment to activate only a subset of nodes. The coordinated scheduling of the sensor duty cycle [33] increases the probability of detection and reduces the detection delay with a minimal power consumption. Communication protocols turn off the radio when a node is not the intended receiver [12]. Though many individual solutions are proposed, few real systems actually achieve power efficiency comprehensively, which makes the integrated approach in VigilNet novel and practically useful. Considering the diversity of the different approaches, we categorize power management strategies in the context of two types of systems: sampling systems and surveillance systems.

#### A. Power Management in Sampling Systems

Great Duck Island [26] and Structural Monitoring [27] are typical sampling systems, which are deployed as distributed large-scale data acquisition instruments. Power management strategies in these systems normally make use of the following techniques:

- **Predefined sampling schedules:** Most environmental phenomena, such as temperature, exist ubiquitously over space and continuously over time. The static nature of these phenomena makes it sufficient to construct the data profile by sampling the environment within discrete time and space. Nodes can conserve energy by turning themselves off, according to a predefined schedule.
- **Synchronized and coordinated operations:** Once the sampling interval is defined *a priori*, nodes can communicate in a synchronized fashion. With a precise time synchronization [31], a receiver can turn on the radio module right before the message payload arrives. Consequently, we can avoid low-power listening over radio [10] during a non-active period. In addition, with the knowledge about the sending rate of individual nodes, we are able to estimate the link quality without control messages [34].
- **Data aggregation and compression:** Since channel media access is costly, especially when the receiver is in a deep-sleep state [10], it is beneficial to send out one aggregate containing multiple sensing readings [19][20]. In addition, due to the value locality of the sensed data, we can compress the total number of bits to be sent over the air. Since both aggregation and compression need to buffer a relatively large number of readings, which introduces a certain delay, they are not quite suitable for time-critical surveillance systems. However, they match most sampling systems very well.

#### B. Power Management in Surveillance System

On the other hand, operations in surveillance systems [35][36][37] [38], such as VigilNet, are event-driven in nature. In surveillance systems, we are more interested in the data profile between inception and conclusion of the transient events. These systems should remain dormant in the absence of the events of interests, and switch to an active state to obtain high fidelity in detection. Normally, the surveillance systems improve the system lifetime through the following approaches:

- **Coverage control:** Surveillance systems are normally deployed with a high density (For instance, the default configuration of VigilNet [38] has 28 nodes per nominal radio range (30 m)) for the sake of robustness in detection and fine-grained sensing during tracking. We can increase the system lifetime by activating only a subset of nodes at a given point of time, waiting for potential targets.
- **Duty cycle scheduling:** The duration of transient events within the area of surveillance is normally non-negligible. By coordinating nodes' sleep schedules, we can conserve energy without noticeably reducing the chance of detection. Duty cycle scheduling is different from sample scheduling in the sense that duty cycle scheduling is at the micro-scale (milliseconds vs. minutes) and it is strongly affected by the dynamics of the events (e.g., target velocity).
- **Incremental activation:** The sampling systems are normally designed for data logging. At each sample instance, all sensors should be activated to obtain a complete data profile. In contrast, surveillance systems are designed to detect transient events of interest. It is sufficient to activate only a subset of sensors for the initial detection. After the initial detection, we can activate other sensors to achieve a higher sensing fidelity and to perform classification.

### III. POWER MANAGEMENT REQUIREMENTS IN VIGILNET

Our power management strategies are motivated by a typical military surveillance application. The mission objective of such a system is to conduct remote, persistent, clandestine surveillance to a certain geographic region to acquire and verify enemy capabilities and transmit summarized intelligence worldwide in a near-real time manner. Several system requirements affect our power management design within VigilNet:

- **Continuous surveillance:** Due to the dynamic/transient nature of the event, VigilNet is required to provide continuous surveillance. This requirement significantly affects the overall architecture of power management strategies and the degree of energy conservation VigilNet can achieve.
- **Real-time:** As a real-time online system for target tracking, VigilNet is required to cope with fast changing events in a responsive manner. The delays introduced by the power management directly affect the maximum target speed our VigilNet can track. It is an essential

design tradeoff to balance between network longevity and responsiveness.

- **Rare and critical event detection:** Due to the nature of military surveillance, VigilNet deals with the rare event model. In this model, the total duration of events is small, compared to the overall system lifetime. On the other hand, events are so critical that the power management becomes a secondary consideration in the presence of events.
- **Stealthiness:** Deployed in hostile environments, it is vital for VigilNet to have a very low profile. Miniaturization makes nodes hard to detect, physically; however, radio messages can be easily intercepted if nodes frequently communicate. Power management protocols designed for VigilNet should maintain silence during surveillance in the absence of significant events.
- **Flexibility:** We envision the deployment of VigilNet under different densities, topologies, sensing and communication capabilities. Therefore, it is essential to design a power management architecture that is flexible enough to accommodate various system scenarios.

#### IV. KEY POWER MANAGEMENT STRATEGIES IN VIGILNET

In order to achieve long-term surveillance that meets the military requirement (e.g., 3~6 months), an aggressive 12~24 fold life-time extension is essential. Our initial investigation [38] indicates that a single power management strategy is neither sufficient nor flexible. Therefore we restructure our prototype system described in [38] by adding a new combination of tripwire service and duty cycle scheduling. We believe this is the right direction to pursue. In this section, we detail three main strategies, namely the tripwire service, sentry service and duty cycle scheduling, before presenting an overarching architecture in the next section. In order to support these strategies, all nodes within VigilNet find their positions with an accuracy of 1~2 meters and they synchronize with each other within 1~10 milliseconds using the techniques described in [39] and [31], respectively. Long-range communication devices are deployed as bases to relay sensor reports outside of the sensor field.

##### A. Tripwire Services

This section proposes a novel network-wide power management strategy called *Tripwire Service*. This service divides the sensor field into multiple sections, called *tripwire sections*, and applies different working schedules to each tripwire section. A tripwire section can be either in an active or a dormant state, at a given point of time. When a tripwire section is dormant, all nodes within this section are put into a deep-sleep state to save energy. Surveillance in active tripwire sections can be done by either turning all nodes on or applying coverage algorithms such as the sentry service discussed later in Section IV-B. The rationale behind the tripwire service is the existence of roads in the area of interest. By deploying the tripwire along the road, we can guarantee the detection without activating all sensors in the area.

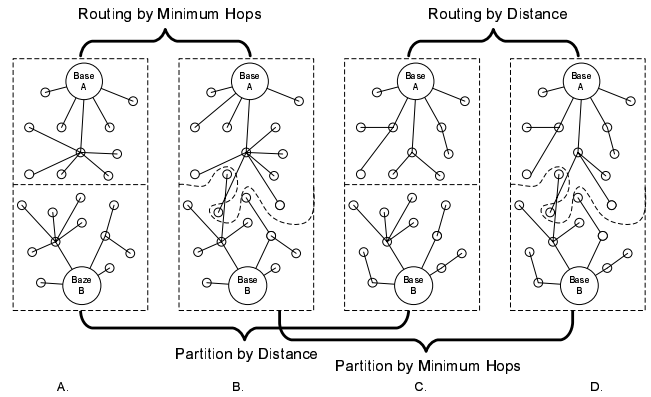


Fig. 1. Four Different Partition Methods

1) *Tripwire partition:* VigilNet implements its tripwire partition policy based on the Voronoi diagram. A network with  $n$  bases is partitioned into  $n$  tripwire sections such that each tripwire section contains exactly one base  $i$  and every node in that tripwire section is closer to its base  $i$  than to any other base inside the sensor field. Every node in the network uniquely belongs to one and only one tripwire section. The rationale behind Voronoi partition is to reduce the energy consumption and the end-to-end delay in data delivery.

The positions of bases directly determine the layout of tripwire sections and affect the routing path length for individual nodes. The optimal base placement method to minimize the average path length to the nearest base can be found at [40]. In practice, the base placement strategy is normally determined by the mission plan and topology.

2) *Tripwire partition mechanism:* This section describes the mechanism to enforce the tripwire partition policy. At the beginning of the tripwire partition operation, each base broadcasts one initialization beacon, to its neighbors with a hop-count parameter initialized to one. Link Symmetry Detection [41] is used to ensure beacons can only be received through high quality symmetric links. Each receiving node maintains the minimum hop-count value of all beacons it received from the nearest base, in terms of the physical distance, and ignores beacons with higher hop-count values and those beacons from other bases. Beacons are flooded outward with hop-count values incremented at every intermediate hop. Through this mechanism, all nodes in the network get the shortest high quality path, in *hops*, to the nearest base, in *physical distance*. While the above mechanism is intuitive, the design deserves some further clarification. First, the boundaries between partitions are well delimited if we partition the network according to the physical distance between sensor nodes and bases (Figure 1A and 1C). If the communication hop is used instead, the radio irregularity and the interference cause partitions to interleave with each other (Figure 1B and 1D). This brings complexity and uncertainty to the design of optimal tripwire placement strategies. Second, it is beneficial to use hop counts to build diffusion trees within each partition, because 1) the normal geographic-based routing does not guarantee high-quality shortest path to the root. 2) Due to the existence of

high-quality long links, a smaller number of nodes become active backbone nodes in the hop-based routing than in the geographic-based routing. Finally, this design provides certain robustness to the base failure. If a base fails, the sensor field can be easily repartitioned without this base.

3) *Tripwire scheduling*: A tripwire section can be either in an active or a dormant state. We configure the state of each tripwire section by setting a 16 bits schedule at the corresponding base. Each bit in the schedule denotes the state of this tripwire section in each round (rotation) up to 16 rounds. After 16 rounds, the pattern is repeated. With this design, we can assign  $65536$  different schedules to each tripwire and assign  $65536^N$  ( $N$  is the number of tripwires.) different schedules to the network. The schedule can be predetermined or randomly generated. Random scheduling is done by setting the Tripwire Duty Cycle (TDC), which is the percentage of active rounds in the schedule.

### B. Sentry Services

In order to exploit the high node density within the sections, we design and implement a section-wide power management strategy, called *sentry service*. The main purpose of the sentry service is to select a subset of nodes, which we define as *sentries*, in charge of surveillance. Sentry selection contains two phases. Nodes first exchange neighboring information through hello messages. In each hello message, a sender attaches its node ID, position, number of neighbors and its own energy readings. After the first phase, each node builds up a one-hop neighbor table. In the second phase, each node sets a delay timer. The duration of the timer is calculated based on the weighted Energy rank  $R_{energy}$  and weighted Cover rank  $R_{cover}$  as shown in Equation 1. The energy rank  $R_{energy}$  is assigned according to energy readings among neighboring nodes (e.g., the node with the highest energy reading within a neighborhood has a rank of 1.) Similarly, the cover rank  $R_{cover}$  is assigned according to the number of neighbors within a node's sensing range. As for current implementation, we assign equal weights to both ranks.

$$T_{timer} = \frac{W_e \times R_{energy} + W_c \times R_{cover}}{(W_e + W_c) \times \#Neighbors} MaxDelay + Jitter \quad (1)$$

After the delay timer fires in one node, this node announces itself as *sentry* by sending out a declaration message. While other nodes, in the vicinity of the declaring node, cancel their timers and become dormant *non-sentry* nodes. The effective range, in physical distance, of a sentry's declaration message is defined as the Range of Vicinity (ROV). While the sentry selection can be straightforwardly implemented, the challenging part is to *choose* and to *enforce* the appropriate range of vicinity (ROV). This parameter directly affects the sentry density, hence affects the lifetime of the network.

1) *How to choose ROV*: The appropriate ROV value can be chosen by the analytical intrusion detection model detailed in Appendix. This model describes the relationship between the detection probability, the sensing range and the sentry density.

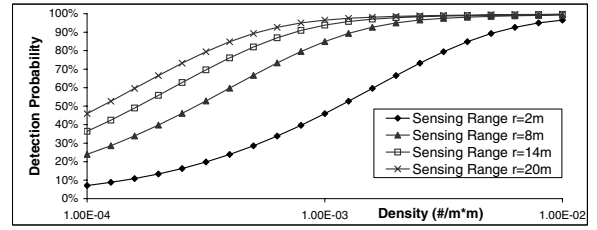


Fig. 2. Detection Prob. Vs. Sentry Density

Since, theoretically, there is at most one sentry within each ROV range, according to the circle covering theorem [42], the sentry density is upper bounded by  $\frac{2\pi}{\sqrt{27}ROV^2}$ . Given the area size, sensing range and sentry density, we get the detection probability (Figures 2) according to the derived model. For a typical deployment with 1000 nodes in  $100 \times 1000$  m<sup>2</sup> area, Figure 2 indicates how to choose the right combination of system configurations. For example, in order to achieve a 99% detection probability, we can choose either a sentry density of 0.008 nodes/m<sup>2</sup> (ROV= 6 meters) with 8 meter sensing range or a lower density of 0.004 nodes/m<sup>2</sup> (ROV=8.5 meters) with 14 meters sensing range. We note that when ROV is set to the sensing range, we can guarantee 100% detection, assuming no voids.

2) *How to enforce ROV*: After we choose a ROV value, we need to enforce it during the sentry selection phase. Since the sensing range is normally smaller than the radio range, directly using the radio range as the ROV cannot guarantee an effective coverage of the area. For example, the HMC1002 dual-axis magnetometer used by MICA2 has only 30-feet effective range for a moving car. If we use the Chipcon radio (>100 feet) to define the range of vicinity, less than 10% of area is sensing covered. There are two approaches to address these issues. The first approach is to reduce the radio sending power to emulate the ROV range. The power setting can be chosen in such a way that there is about one sentry within each sensing range. The second approach is to discard declaration messages from any sentry beyond the distance of ROV. The first approach achieves sensing coverage, without the location information of the nodes [43], while the second approach provides a more predictable sentry distribution, because the emulated ROV would be affected by the radio irregularity in the environment. Consequently, we adopt the second solution in our system, given the fact that localization [39] is supported in VigilNet.

### C. Sentry Duty Cycle Scheduling

The requirement for continuous sensing coverage in the sentry service imposes a theoretical upper bound on the system lifetime. This upper bound is decided by the total number of nodes deployed. Since a target normally stays in the sensing area of a sentry node for a non-negligible period of time, it is not necessary to turn sentry nodes on all the time. By using duty cycle scheduling, we are able to break the theoretical upperbound imposed by the full coverage algorithms [7]. Let  $T_{on}$  be the active duration and  $T_{off}$  be the inactive duration, then

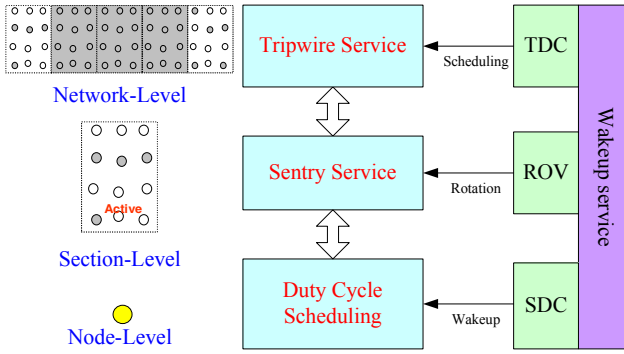


Fig. 3. Integrated PM Architecture

the Sentry Toggle Period (STP) is defined as  $(T_{\text{on}} + T_{\text{off}})$ , and the Sentry Duty Cycle (SDC) is defined as  $\frac{T_{\text{on}}}{STP}$ . Theoretically, the duty cycle scheduling can achieve unbounded energy conservation by lowering the SDC value. The paramount concern of this technique is that lowering the SDC value increases the detection delay and reduces the detection probability. We can either effectively implement random duty cycle scheduling or more sophisticated scheduling algorithms to coordinate node activities to maximize performance. In [33], we demonstrate a local optimal scheduling coordination algorithm to reduce the detection delay and increase the detection probability. We prove that, at relatively large SDC (e.g.  $5\% < SDC$ ), the difference between random scheduling and optimal scheduling can be practically ignored. Since the random scheduling does not need control messages for coordination (more stealthy) and it is not affected by time drift, we choose random scheduling over the coordinated one in the system implementation.

## V. INTEGRATED SOLUTION: TRIPWIRE-BASED POWER MANAGEMENT WITH SENTRY SCHEDULING

To achieve an aggressive network lifetime extension, the VigilNet power management subsystem integrates the three strategies mentioned in previous sections into a multi-level architecture, as shown in Figure 3. At the top level, the tripwire service controls the network-wide distribution of power consumption among sections; the uniform discharge of energy across sections is achieved through the scheduling mechanism we discussed in section IV-A.3. We use a Tripwire Duty Cycle (TDC), which is the percentage of active time for each tripwire section, to control the network-wide energy-burning rate. There are two special cases: when TDC equals 100%, the whole network becomes active and the tripwire service is merely a network partition service. When TDC equals 0%, the whole network is in dormant status and it can only be awoken by external sources. At the second level, the sentry service controls the power distribution within each section. The uniform discharge of energy in a section is achieved through automatic rotation strategies according to the remaining power within individual nodes. We use the Range of Vicinity (ROV) parameter to control the energy-burning rate of active sections. When ROV equals the sensing range of nodes, the section is fully covered. A higher ROV value

than the sensing range leads to a partial coverage and a lower ROV value than the sensing range leads to redundancy in the coverage. When ROV equals 0 meter, the sentry service is actually disabled and all nodes with the section are awake, providing the highest degree of coverage. At the third-level, duty cycle scheduling controls the energy-burning rate of individual sentry nodes by manipulating their wakeup/sleep schedule. The Sentry Duty Cycle (SDC) parameter is used to control the awareness of sentry nodes, which is the percentage of active time. The duty cycle scheduling can be disabled by setting SDC to 100%. By adopting different values for TDC, ROV and SDC, we can flexibly adjust our power management to accommodate different system scenarios.

## VI. OTHER ENERGY CONSERVATION TECHNIQUES

Besides the three main power management strategies, several other techniques have been integrated into various aspects of the VigilNet system. Similar techniques [20][44][27][10] have been proposed in the literature and we provide this section for the completeness of the VigilNet power management design and implementation.

- **Minimum connected dominating tree:** To ensure a swift delivery of messages, VigilNet requires an active diffusion tree over active tripwire section. Since the communication range is normally much larger than the sensing range [5][2], it is possible to build a diffusion tree on top of sentry nodes. To reduce the energy spent during idle listening, VigilNet desires a tree with the minimum connected dominating set (a tree with minimum non-leaf nodes). Since it is a NP-Complete problem to find the minimum connected dominating set of a graph, we adopt a localized approximation as follows: during the building process, each node rebroadcasts the hop-count beacon after a certain time delay. The delay in one node is inversely proportional to the number of neighbors and the energy remaining. By doing so, the node with more neighbors and more energy left has a higher chance to become the parent node within the diffusion tree.
- **Data aggregation:** The channel media access in wireless sensor network is relatively expensive. For example, in the Chipcon radio implementation for MICA2, to deliver a default payload size of 29 bytes, the total overhead is 17 bytes (37%!), including 8 bytes preamble, 2 bytes synchronization, 5 bytes header and 2 bytes CRC. This motivates us to utilize various kinds of aggregation techniques. The first technique we use is called Application-Independent Aggregation, which concatenate data from different modules into one aggregate, regardless of their semantics. For example, system-wide parameters can be sent with time synchronization messages. The second technique we use is called Application-Dependent Aggregation. The tracking subsystem in VigilNet performs the in-network aggregation by organizing the nodes into groups. Instead of each node reporting its position separately, a leader node calculates the weighted center

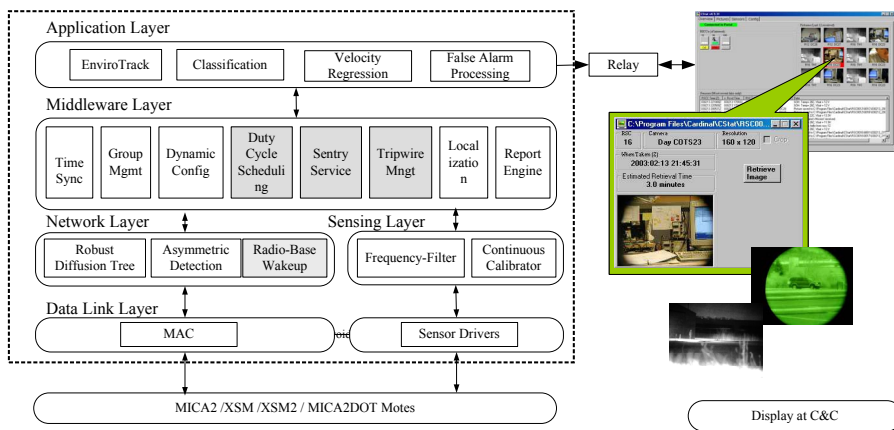


Fig. 4. The VigilNet System Architecture

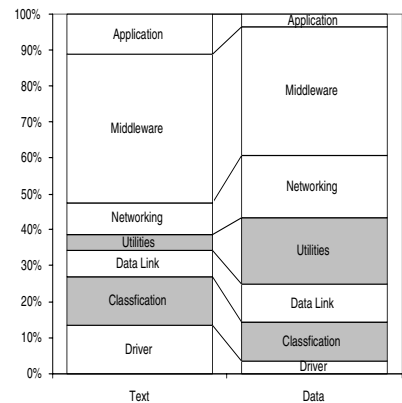


Fig. 5. Memory Layout

of gravity from multiple inputs and reports only one aggregate back to the base.

- Implicit acknowledgement:** Given that the sensor payload is very small, it might not be energy efficient to acknowledge every packet explicitly. Implicit acknowledgement can be achieved through several approaches. They differ in functionality and overhead. B-MAC [10] provides an efficient implementation of the CSMA protocol with radio-layer acknowledgement support. Observing that most of the packets need to be forwarded for routing, we alternatively implemented the acknowledgement as a special field in outgoing packets. When there are no outgoing packets for a period of time, a special acknowledgement packet is sent.
- Incremental detection:** Multi-sensing modalities are desired for achieving target classification. However, it is not necessary to activate all sensors only for detection. Among the three types of sensors in XSM motes, the optic TR230 PIR sensor has the longest detection range and a relatively low power consumption, i.e., 0.88mW. We use this sensor to support the initial detection and to incrementally wakeup other sensors for classification purposes.
- Passive wakeup circuitry :** Several efforts [2][45][32] have been made to support low-power passive wakeup by using an acoustic detector [45], infrared sensor [2] or radio [32]. Currently, the hardware-event-driven design [2] of XSM motes is not mature enough for VigilNet to exploit this aspect. However, this is a very promising direction.

## VII. TRADEOFF: PERFORMANCE VS. ENERGY EFFICIENCY

One key research challenge for VigilNet is to reconcile the need for network longevity with the need for fast and accurate target detection and classification. The former requires most sensor nodes to remain inactive, while the later desires many active sensor nodes. As we mentioned before, the event model directly affects the design of the power management. Energy efficiency can be comparatively easy to achieve if events of interests are ubiquitously present. The data quality of some

events, such as temperature and humidity, are not directly correlated with the responsiveness of the system. While in the surveillance system, *responsiveness* and *awareness* directly affect the system performance including tracking and classification. The former can be measured in terms of the detection probability and delay, and the later can be measured in terms of the number of nodes detecting external events, simultaneously. We have investigated responsiveness in previous sections IV-B and IV-C. This section focuses on how to improve the system awareness. In VigilNet, awareness is supported by the *on-demand wakeup service*. The on-demand control is stealthier compared to the periodic control [38], because wakeup beacons are sent only when events occur. To support the on-demand control, we need to guarantee the delivery of wakeup beacons. Because of the special stealthiness requirement, the non-sentries cannot synchronize their clocks with their sentries by exchanging messages. Therefore, neighboring non-sentry motes may no longer have a sleep-wakeup cycle synchronized with each other due to the clock drift, and a sentry cannot keep track which of its neighbors are awake. To guarantee delivery, a non-sentry periodically wakes up and checks radio activity (detects preamble bytes) once per checking period (e.g., every second). If no radio activity is detected, this node goes back to sleep, otherwise it remains active for a period of time, preparing for incoming targets. If a sentry node wants to wake up all neighboring nodes, it only needs to send out a message with a long preamble with a length equal to or longer than the checking period of non-sentry nodes. Since in the rare event model, the wakeup operations are done very infrequently, the long preamble doesn't introduce much energy consumption in sentry nodes. On the other hand, since the amount of time taken to check the radio activity is constant for a specific radio hardware, the length of checking period determines the energy consumption in non-sentry nodes. In general, a long checking period leads to a lower energy consumption. However, to ensure that a sentry node wakes up neighboring non-sentry nodes before a target moves out of their sensing range, the checking period can not be arbitrarily long. Theoretically, the upperbound of checking period is  $\frac{\sqrt{R^2 - r^2}}{S}$ , where  $R$  is radio



Fig. 6. Location of Deployment and A Deployed XSM Mote

range,  $r$  is sensing range of sentries and  $S$  is the speed of target. Due to the other delays, such as sensor warm-up time, the checking period should be smaller than this theoretical bound. In our implementation, non-sentry nodes have 1% duty cycle with 1 second checking period.

### VIII. IMPLEMENTATION

The power management architecture described in Section V has been integrated into the VigilNet system. We have successfully transferred VigilNet to a military agency for deployment by the end of 2004. The overarching architecture of VigilNet is shown in Figure 4. The components in gray are specially designed for the power management purpose. Other components provide extra energy-aware features, as mentioned in section VI.

VigilNet is built on top of the TinyOS operating system. TinyOS supports a lightweight event driven computation model with two-level scheduling. VigilNet is mostly written in NesC, a derivative language from C specially designed for embedded programming. This language enables the programmers to define the interfaces, functions of components and the relations (dependencies) among them. The size of VigilNet is about 40,000 lines of code, supporting multiple existing mote platforms including MICA2 and XSM. The compiled image occupies 83,963 bytes of code memory and 3,586 bytes of data memory. The code and data memory maps are shown in Figure 5.

We categorize the system components into seven groups; Data link and sensor driver layers use default components in TinyOS; Network layer consists of three major components: robustness diffusion tree, asymmetric link detection [41] and radio-based wakeup service. The sensing layer provides detection and classification with continuous calibrator and frequency filters [1]. We note that it is very critical to have a sensing subsystem with minimal false alarms in an outdoor environment. Otherwise, the network lifetime is severely reduced due to unnecessary wakeup operations. The application layer focuses on tracking and high-level classification [46]. The middleware layer occupies most code (40%) and data memory (35%). Among all the middleware services, the tripwire service, sentry selection, duty cycle scheduling and wakeup

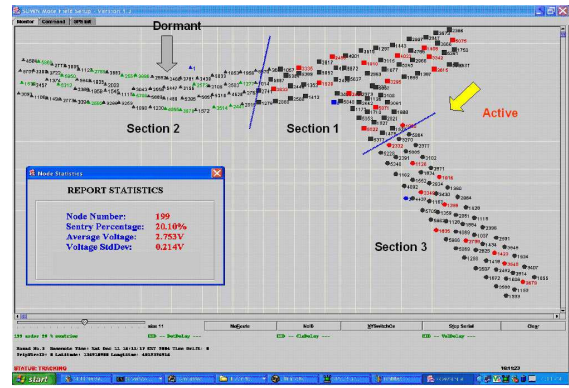


Fig. 7. Tripwire Partition

service form the basis for power management subsystem. Their functionalities are supported by other services. For instances, the localization service provides the basis for the tripwire partition and sentry selection. The group management service allows power-efficient data aggregation. The configuration service facilitates the online tuning of the power management parameters. Multilevel sleep modes in the ATmega128 permit a high-granularity control of power dissipation. Selectable transmission power settings (255 levels) in CC1000 enable us to adjust the effective range of sentry declaration messages dynamically.

### IX. SYSTEM EVALUATION

This section presents experimental results that evaluate the performance of the power management subsystem. The experimental results in Section IX-A are obtained through an actual deployment of 200 XSM motes, focusing on the sentry selection, tripwire partition and tracking delays. Other experiments in Section IX-B, especially those related to the system lifetime, require a significant amount of time. Unfortunately, we currently can not afford to deploy such a large system unattended for a long time. We have to conduct those evaluations through a hybrid approach, which uses basic measurements from a smaller number of motes as input to a simulator. By doing so, we can investigate the impact of different system configurations on the performance of power management.

#### A. Field Evaluation

The field evaluation was done as part of a technical transition on December 2004, when we deployed 200 XSM motes on a dirt T-shape road (200 meters by 300 meters). The XSM mote is designed by the joint efforts of Ohio State University [2] and CrossBow Inc, which features an Atmel ATmega128L microcontroller and a Chipcon 433MHz CC1000 radio. Its sensing suite includes magnetic, acoustic, photo, temperature and passive infrared sensors (PIR). Figure 6 displays the environment where our system was located and the picture of one of the XSM motes. Nodes are randomly placed roughly 10 meters apart, covering one 300-meter road and one 200-meter road.

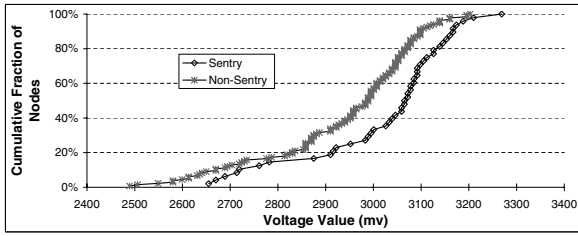


Fig. 8. Effectiveness of Sentry Selection

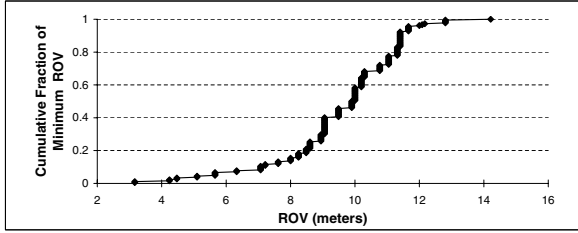


Fig. 9. ROV Enforce Results

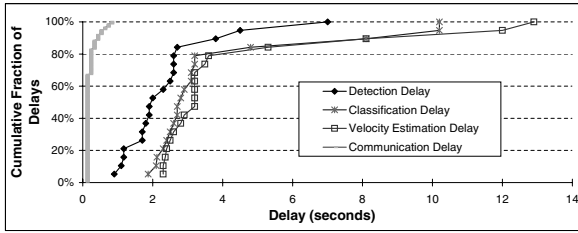


Fig. 10. Distribution of Different Delays

1) *Effectiveness of the tripwire partition:* One snapshot of the network layout collected by our GUI is shown in Figure 7. We placed 200 XSM field motes and 3 mica2dot base motes in the field. Accordingly, the network is divided into three sections. The layout indicates that the Voronoi-based tripwire partitioning is very effective and that all nodes attach to the nearest base nodes through the shortest path.

2) *Effectiveness of the sentry selection:* In this experiment, we evaluate the effectiveness of sentry selection. Figure 8 plots the cumulative distribution function of the voltages of nodes within the network. The left curve is the voltage CDF of non-sentry nodes and the right curve is the voltage CDF for sentry nodes. It confirms that our sentry selection process is effective and that nodes with high remaining energy have a high probability to be chosen as sentries. For instance, none of nodes with voltage below 2.65V is chosen as a sentry. Figure 8 further confirms that it is not the case that nodes with high voltages are always selected as sentries, due to the random jitter introduced in Equation 1 and to the localized selection process on a non-uniform distribution of XSM motes.

3) *Effectiveness of ROV enforcement:* We also investigate the effectiveness of enforcing Range of Vicinity (ROV), when we set the system parameter ROV as 10 meters. Figure 9 shows the cumulative distribution function of minimum distances between sentry-pairs. The average minimum is 9.57 meters with 1.88 meters standard deviation. We note that due to

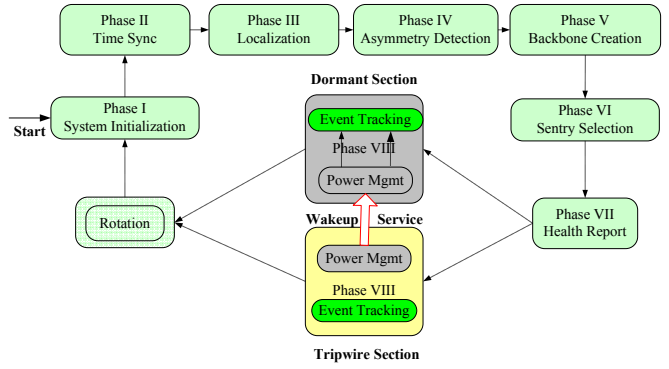


Fig. 11. Phase Transition and Rotation

TABLE I  
POWER CONSUMPTION ACCORDING TO THE MOTE STATE.

Node state	Radio State (Messages per second)	Processor State	Sensors State	Total Power
Init	receive (2)	active	off	49.449mW
SentrySleep	off (0)	sleep	off	42μW
NonSentrySleep	LPL (0)	sleep	off	450μW
AwakeComm	receive (2)	active	off	49.449mW
AwakeCommSensing	receive (2)	active	on	71.45mW
AwakeSensing	receive (0)	active	on	70.01mW

the radio irregularity introduced by the ground effect in the outdoor environments, a small percentage of sentry nodes can not reach each other, even when they are very close (<5 meters) to each other.

4) *Delays under power management:* In this experiment, we investigate various delays under power management. When a target enters the surveillance area, a detection report is issued first, followed by classification reports. Finally, after sufficient information is gathered, velocity reports are issued. Figure 10 illustrates the cumulative distribution of different delays. The communication delay (leftmost curve) is much smaller compared with other delays. About 80% of detections are done within 2 seconds. Over 80% of the classification and velocity estimations are made within 4 seconds. This empirical result indicates that our power management does not degrade the tracking performance significantly.

TABLE II  
KEY SYSTEM PARAMETERS

Parameter	Definition	Default Value
SDC	Sentry duty cycle (see IV-C)	25%
STP	Sentry toggle period (see IV-C)	1 second
SSA	Sentry service activation	True
TN	Number of tripwire partitions in the network	1
TDC	Tripwire duty cycle percentage (see IV-A.3)	100%
VS	Target Speed	4 m/s
RN	Number of system rotations per day	1
SR	Sensing Range	10 m
RR	Radio Range	30 m

## B. Hybrid Evaluation

In the hybrid evaluation, we use the experimental measurements from the XSM platform as inputs to a discrete event



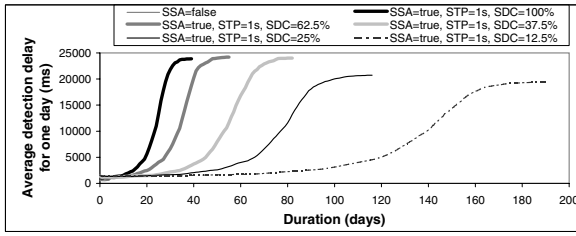


Fig. 12. Influence of sentry duty cycle (SDC) on average detection delay(ADD).

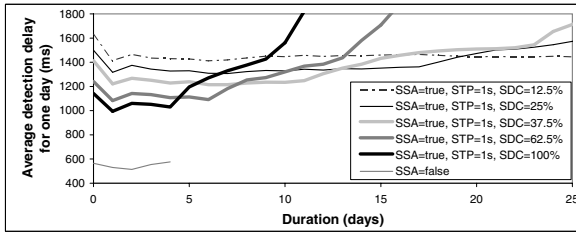


Fig. 13. Influence of sentry duty cycle (SDC) on average detection delay (ADD) (the second view)

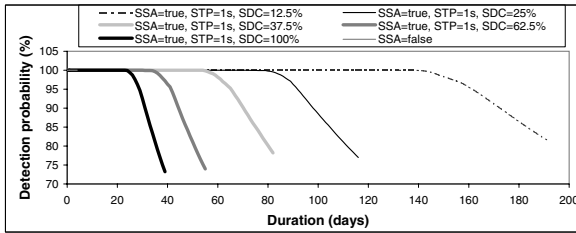


Fig. 14. Influence of sentry duty cycle (SDC) on detection probability (DP).

simulator. This simulator emulates the multi-phase VigilNet operations as shown in Figure 11. We distribute 10,000 nodes randomly within a 1,000,000 m<sup>2</sup> square. VigilNet initializes in three minutes with a sequence of phases (from Phase I to VII). After that, VigilNet enters the surveillance phase (Phase VIII). The system rotates periodically to refresh system-wide soft-states and balance the power consumption. The number of rotations per day is defined as  $RN$  as shown in Table II. A target enters the network randomly at one of the edges and exits randomly at the opposite edge of the area. To emulate the sensing delay we experienced in the real testbed, we consider that a target is detected when it is within the sensing range of an active node for at least 5 milliseconds and when that node can reach its tripwire base station to report the event.

1) *Battery model:* We obtained similar empirical power consumption results as reported in [2], which provides very complete analysis of XSM motes. XSM motes use two standard AA (A91) batteries. Each battery has an energy capacity uniformly chosen between 2,848mAh and 2,852mAh [47]. However, to model reality better [48], we suppose that a mote dies when it has used 85% of the available energy.

The sensor nodes are in one of six power consumption states at any time. We list and detail the power consumption of these

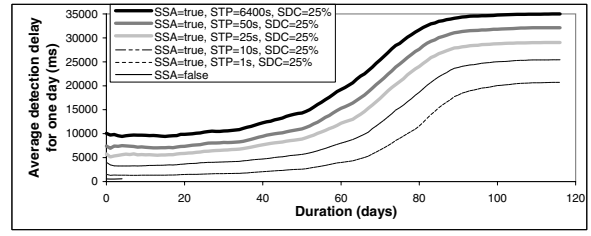


Fig. 15. Influence of sentry toggle period (STP) on average detection delay (ADD).

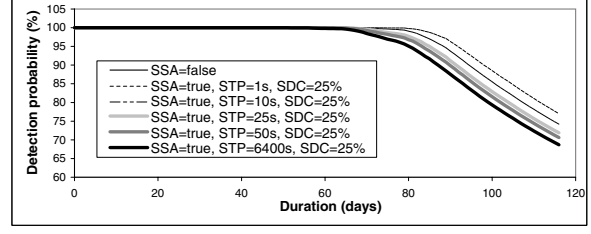


Fig. 16. Influence of sentry toggle period (STP) on detection delay (DP).

six states in Table I. When a message is transmitted, the radio switches to the transmit state for 30ms (a typical time required by XSM nodes to send a message under the MAC contention). The indicated number of messages per second in Table I is an upper bound result from the empirical observations.

2) *Performance metrics and system parameters:* We investigate three major performance metrics under different system configurations. 1) Detection Probability (**DP**), which is the percentage of successful detections among all targets that enter into the system during one day. 2) Average detection Delay (**ADD**), which is the average time elapsed between the entrance of a target into the area and its detection by one of sensor nodes. 3) Network lifetime (**NL**), which is defined as the number of days for which the detection probability of a target remains greater than 90%. The key system parameters are listed in Table II. Unless mentioned otherwise, the default values in Table II are used in all experiments. The baseline for comparison is VigilNet without any power management.

3) *Impact of the sentry service and duty cycle scheduling:* In this section, we evaluate the energy savings achieved by the sentry service and the duty cycle scheduling. In particular, we study the influence of the activation of the sentry service (SSA), of the sentry duty cycle (SDC), and of the sentry toggle period (STP) on energy consumption. One hundred targets are simulated during each rotation to obtain statistics in detection probability, but we take into consideration the power consumption of only ten of them (VN=10) as the real workload. As previously mentioned, we use a network of 10,000 nodes randomly distributed within a square of 1km edge length. Each node has a radio range of 30 meters. This configuration matches our real system requirements dictated by the military: nodes have an average of 27.5 neighbors within their communication range, and an average of 3.1 neighbors within their sensing range.

Figures 12, 13 and 14 show the variations of the average

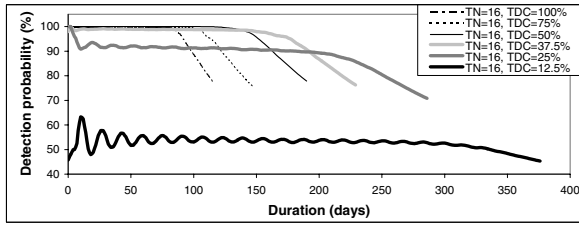


Fig. 17. Influence of tripwire duty cycle (TDC) on detection probability (DP).

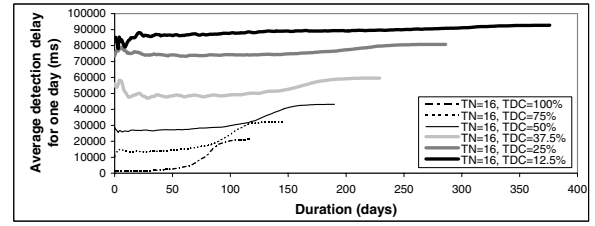


Fig. 18. Influence of tripwire duty cycle (TDC) on average detection delay (ADD).

detection delay, detection probability, and network lifetime, according to the sentry duty cycle. Figure 13 takes a closer look at a particular section of Figure 12. We first observe that without the sentry service (SSA = false), the lifetime of the network (NL) is short: all the nodes run out of energy after only four days. The activation of the sentry service increases the lifetime of the network by approximately seven times. However, it also slightly increases the average detection delay of a target by 1 second. This could be expected as the first nodes that the target encounters may be dormant. We note that the delay is relatively small (e.g., 1~2 seconds), as shown in Figure 13.

The use of duty cycle scheduling ( $SDC \neq 100\%$ ) significantly improves the network lifetime. For instance, with a duty cycle of 12.5%, the lifetime of the network is multiplied by about five times. This may be surprising: we would expect the network lifetime when  $SDC=12.5\%$  to be approximately eight times the network lifetime when  $SDC=100\%$ . The observed values are due to the energy consumed during the rotation phase and when target detection occurs. These tasks consume a non-negligible amount of energy and therefore impose a limit on the network lifetime.

We remark that during the first four days of network operation, the average detection delay is shorter when the sentry duty cycle is higher. This could be expected when a target enters the sensing range of a sentry node, this node may be in a dormant state. We note that the difference between the average detection delays for different values of SDC (Figure 13) is no more than one second. This can be explained by the short sentry toggle period (1 second).

Figure 14 shows the influence of the sentry duty cycle (SDC) on the detection probability. We observe that, for all configurations, the initial detection probability is 100%. As nodes start to run out of power, the detection probability decreases until all the nodes become dysfunctional. On average, during the network lifetime, the successful detections reported in Figure 13 occur between 0.5 second and 2 second after the target entered the square area. Beyond the network lifetime, VigilNet can still detect the targets, however the delay increase gradually as shown in Figure 12.

In Figure 15 and Figure 16, we study the effect of the sentry toggle period (STP) on the average detection delay and the detection probability. We fix the sentry duty cycle at 25%. We observe that a greater toggle period negatively impacts the average detection delay. Indeed, if the toggle period is small

(e.g., 1 second), a dormant sentry, having a target entering its sensing range, wakes up with a high probability before this target exits the sensing range. Conversely, if the toggle period is big (e.g., 6400 second), a dormant sentry has a low probability of being waken up before the target leaves its sensing range.

**Guidelines:** From the analysis of this section, we can conclude the following. First, to reduce the detection delay, we must choose a sentry toggle period as small as possible. Second, to increase the network lifetime, we advise to select a small sentry duty cycle. However, note that the time during which a sentry remains awake cannot be arbitrarily small, because it is limited by the time necessary to warm up the sensors and by the time necessary to gather enough sensor data to infer whether there is a target or not. Consequently, rapid sensor wake up and quick target detection algorithms are features that can significantly extend the lifetime of a sensor network. Efforts in this direction are worthwhile.

4) *Impact of the tripwire service:* We investigate both the grid and the random placement of tripwire bases. In the case of a Tripwire Number (TN) greater than 16, the two placement strategies generate similar results. For TN smaller than 16, the grid topology performs better. Due to space constraints, we report here only the results concerning the grid tripwire topology.

We configure the wireless sensor network as in Table II. In Figures 17 and 18, we study the impact of the tripwire duty cycle on the performance of the network. We use sixteen tripwires. As we would expect, the smaller the tripwire duty cycle, the longer the lifetime of the network. For instance, when the tripwire duty cycle equals 25%, the network lifetime is about twice the lifetime obtained when the tripwire duty cycle equals 100%. One could expect a multiplication of the lifetime by four times but this would not take into consideration the energy consumed during the rotation phase and when target detection occurs. Additionally, we observe that the average detection delay is significantly longer when the tripwire duty cycle is small. Indeed, when the tripwire duty cycle is small, a relatively small portion of the network is awake at any given time, and the target may cover a bigger part of the network without being detected. Finally, we notice that a tripwire detection cycle of less than 25% seriously impacts the detection probability during the first weeks of network operation. This is due to the fact that, with such low levels of tripwire activity, a large part of the network may

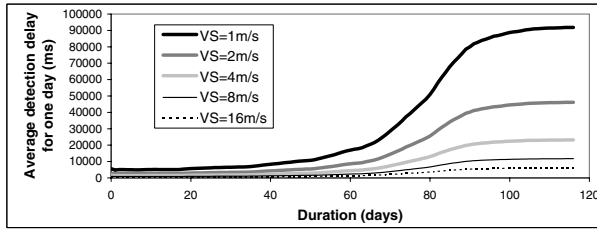


Fig. 19. Influence of target speed (VS) on average detection delay (ADD).

remain dormant for an extended period of time, producing the possibility that the target crosses the network exclusively through such zones.

**Guidelines:** From this section, we can conclude the following: a low tripwire duty cycle increases the network lifetime, but increases noteworthy the detection delay and decreases the detection probability. In a grid deployment, a tripwire duty cycle below 25% is detrimental to the performance. This, on the other hand, indicates the importance of the smart tripwire placement strategy under a low tripwire duty cycle.

5) *Impact of the targets speed:* In this section, we study the effect of the target speed on performance. The configuration of the network is the same as in Table II. Figures 19 and 20 show the influence of target speed on average detection delay and detection probability. We observe that a high target speed decreases the detection delay. This may be surprising as when the target speed increases, it spends less time within the sensing range of a given sensor, thereby decreasing the probability of being detected. However, as the target speed increases, it covers more nodes in a shorter amount of time. The effect of target speed on detection probability is insignificant for  $VS \leq 16\text{m/s}$ . We recall that the sensing range of a sensor is 10 meters in this experiment. At a speed of 16m/s, the target spends a maximum of 1250ms within the sensing range of a given sensor. This duration is sufficient for a sensor to differentiate the target from false alarms.

**Guidelines:** To summarize the results from this section, we can conclude the following. First, it takes more time to detect slow targets than faster ones. Second, a network with characteristics similar to the one defined in this experiment can handle targets with speeds typical of moving terrestrial objects (up to at least 16 meters per second i.e. 35.8 miles per hour).

## X. CONCLUSION

This paper presents a recent major effort to address the energy efficiency for outdoor long-term surveillance. It is a comprehensive case study on power management in a realistic environment with a large testbed. We investigate the power management at the network, section and node level by using a novel tripwire service, sentry service and duty cycle scheduling, respectively. We invest a significant amount of effort to validate our system with a network of 200 XSM motes in an outdoor environment, an extensive simulation with 10,000 nodes, as well as an analytical probabilistic model.

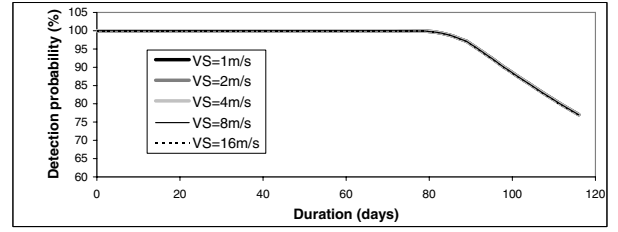


Fig. 20. Influence of target speed (VS) on detection probability (DP).

These demonstrate the effectiveness of our approach and identify several useful guidelines and lessons for the future development of energy-efficient sensor systems.

## XI. ACKNOWLEDGEMENTS

This work was supported in part by the DAPRPA IXO offices under the NEST project (grant number F336615-01-C-1905), the MURI award N00014-01-1-0576 from ONR and NSF grant CCR-0098269. The authors specially thank the NEST program manager Vijay Raghavan for his valuable contributions.

## APPENDIX

Figure 21A shows a rectangular deployment area with sides  $a$  and  $b$ .  $N$  sensor nodes are uniformly deployed in the area, therefore, the node density  $d$  is  $N/ab$ . Assuming that the area is considerably large, therefore, the number of nodes in an area of  $A$  ( $A \ll ab$ ) can be approximated by a Poisson distribution with parameter  $\lambda = dA$ . We assume that the entry point of the intruding target (intruder) is uniformly distributed along all sides, and to make the problem tractable, we assume that the intruder moves along a straight line. The angle between the target direction and the side where the entry point is located is  $\theta$ , which is also considered uniformly distributed and  $\theta \in [0, \pi]$ . The whole picture of the intruding scenario is shown in Figure 21A. Observe that the area of nodes that can detect the intruder contains all points whose distances to the intruder's locus are no larger than sensing range  $r$ . If the length of the intruder's locus in the deployment area is  $L$ , the detection area can be approximated by  $2Lr$ , without considering the edge effect. Based on Poisson distribution, the probability that there is at least one node in this area is  $1 - e^{-2Lrd}$ .

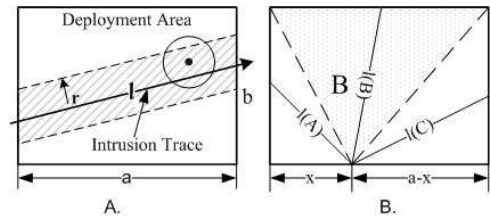


Fig. 21. Intrusion Model

As shown in Figure 21B, the deployment area is divided into three regions. The length of the intrusion trace is:

$$L(\theta, x) = \begin{cases} x / \cos \theta & \text{Locus} \in A \\ b / \sin \theta & \text{Locus} \in B \\ (x - a) / \cos \theta & \text{Locus} \in C \end{cases} \quad (2)$$

We can calculate the expected detection probability  $Expected(P_{detection})$  that an intruder is detected by at least one node by integrating over all entry points on the four adjacent sides of the area.

$$Expected(P_{detection}) = 1 - \frac{F(a, b, r, d) + F(b, a, r, d)}{\pi(a + b)} \quad (3)$$

where  $F(m, n, r, d) =$

$$\int_0^m \left[ \int_0^{\arctan(\frac{n}{x})} e^{-\frac{2rx}{\cos\theta}} d\theta + \int_{\arctan(\frac{n}{x})}^{\pi - \arctan(\frac{n}{x})} e^{-\frac{2rn}{\sin\theta}} d\theta + \int_{\pi - \arctan(\frac{n}{x})}^{\pi} e^{-\frac{2r(x-m)d}{\cos\theta}} d\theta \right] dx \quad (4)$$

## REFERENCES

- [1] L. Gu, D. Jia, P. Vicaire, T. Yan, L. Luo, A. Tirumala, Q. Cao, J. A. S. T. He, T. Abdelzaher, and B. Krogh, "Lightweight Detection and Classification for Wireless Sensor Networks in Realistic Environments," in *SenSys 2005*, 2005.
- [2] P. Dutta, M. Grimmer, A. Arora, S. Biby, and D. Culler, "Design of a Wireless Sensor Network Platform for Detecting Rare, Random, and Ephemeral Events," in *IPSN'05*, 2005.
- [3] J. A. Paradiso and T. Starner, "Energy Scavenging for Mobile and Wireless Electronics," *IEEE Pervasive Computing*, vol. 4, no. 1, 2005.
- [4] S. Roundy, P. K. Wright, and J. Rabaey, "a Study of Low Level Vibrations as a Power Source for Wireless Sensor Nodes," *Computer Communications*, vol. 26, no. 11, 2003.
- [5] *Mica2 data sheet*, CrossBow, available at <http://www.xbow.com>.
- [6] X. Wang, G. Xing, Y. Zhang, C. Lu, R. Pless, and C. Gill, "Integrated Coverage and Connectivity Configuration in Wireless Sensor Networks," in *SenSys 2003*, November 2003.
- [7] T. Yan, T. He, and J. Stankovic, "Differentiated Surveillance Service for Sensor Networks," in *SenSys 2003*, November 2003.
- [8] M. L. Sichitiu, "Cross-layer scheduling for power efficiency in wireless sensor networks," in *IEEE Infocom*, march 2004.
- [9] M. Cardei, M. Thai, and W. Wu, "Energy-efficient target coverage in wireless sensor networks," in *IEEE Infocom*, march 2005.
- [10] J. Polastre and D. Culler, "Versatile Low Power Media Access for Wireless Sensor Networks," in *SenSys 2004*, November 2004.
- [11] T. van Dam and K. Langendoen, "An Adaptive Energy-Efficient MAC Protocol for Wireless Sensor Networks," in *SenSys 2003*, November 2003.
- [12] W. Ye, J. Heidemann, and D. Estrin, "An Energy-Efficient MAC Protocol for Wireless Sensor Networks," in *INFOCOM*, 2002.
- [13] K. Seada, M. Zuniga, A. Helmy, and B. Krishnamachari, "Energy Efficient Forwarding Strategies for Geographic Routing," in *SenSys 2004*, November 2004.
- [14] Y. Xu, J. Heidemann, and D. Estrin, "Geography-informed Energy Conservation for Ad Hoc Routing," in *MobiCom*, 2001.
- [15] T. He, J. Stankovic, C. Lu, and T. Abdelzaher, "SPEED: A Stateless Protocol for Real-Time Communication in Ad Hoc Sensor Networks," in *ICDCS'03*, May 2003.
- [16] M. Agarwal, J. H. Cho, L. Gao, and J. Wu, "Energy-efficient broadcast in wireless ad hoc networks with hitch-hiking," in *IEEE Infocom*, march 2004.
- [17] Y. Yu, B. Krishnamachari, and V. K. Prasanna, "Energy-Latency Trade-offs for Data Gathering in Wireless Sensor Networks," in *IEEE INFOCOM*, 2004.
- [18] W. Choi and S. Das, "A novel framework for energy-conserving data gathering in wireless sensor networks," in *IEEE Infocom*, march 2005.
- [19] S. Madden, M. Franklin, J. Hellerstein, and W. Hong, "TAG: A Tiny Aggregation Service for Ad-Hoc Sensor Networks," in *OSDI'02*, December 2002.
- [20] N. Shrivastava, C. Buragohain, S. Suri, and D. Agrawal, "Medians and Beyond: New Aggregation Techniques for Sensor Networks," in *SenSys 2004*, November 2004.
- [21] S. Bhattacharya, H. Kim, S. Prabh, and T. Abdelzaher, "Energy-Conserving Data Placement and Asynchronous Multicast in Wireless Sensor Networks," in *MobiSys'03*, May 2003.
- [22] B. Chen, K. Jamieson, H. Balakrishnan, and R. Morris, "Span: An Energy-Efficient Coordination Algorithm for Topology Maintenance in Ad Hoc Wireless Networks," in *MOBICOM'01*, 2001.
- [23] W. Heinzelman, A. Chandrakasan, and H. Balakrishnan, "Energy-Efficient Communication Protocol for Wireless Microsensor Networks," in *HICSS*, 2000.
- [24] D. Ganesan, R. Cristescu, and B. Beresford-Lozano, "Power-Efficient Sensor Placement and Transmission Structure for Data Gathering under Distortion Constraints," in *IPSN'04*, 2004.
- [25] A. Bogdanov, E. Maneva, and S. Riesenfeld, "Power-aware base station positioning for sensor networks," in *IEEE Infocom*, march 2004.
- [26] R. Szwedczyk, A. Mainwaring, J. Anderson, and D. Culler, "An Analysis of a Large Scale Habit Monitoring Application," in *SenSys 2004*, November 2004.
- [27] N. Xu, S. Rangwala, K. K. Chintalapudi, D. Ganesan, A. Broad, R. Govindan, and D. Estrin, "A Wireless Sensor Network for Structural Monitoring," in *SenSys 2004*, November 2004.
- [28] J. M. Kahn, R. H. Katz, and K. S. J. Pister, "Next Century Challenges: Mobile Networking for Smart Dust," in *MOBICOM'99*, August 1999.
- [29] K. Kar, A. Krishnamurthy, and N. Jaggi, "Dynamic node activation in networks of rechargeable sensors," in *IEEE Infocom*, march 2005.
- [30] A. Keshavarzian, E. Uysal-Biyikoglu, F. Herrmann, and A. Manjeshwar, "Energy-efficient link assessment in wireless sensor networks," in *IEEE Infocom*, march 2004.
- [31] M. Maroti, B. Kusy, G. Simon, and A. Ledeczi, "The Flooding Time Synchronization Protocol," in *SenSys 2004*, November 2004.
- [32] L. Gu and J. A. Stankovic, "Radio-Triggered Wake-Up Capability for Sensor Networks," in *RTAS'04*, 2004.
- [33] Q. Cao, T. Abdelzaher, T. He, and J. Stankovic, "Towards Optimal Sleep Scheduling in Sensor Networks for Rare Event Detection," in *IPSN'05*, 2005.
- [34] A. Woo, T. Tong, and D. Culler, "Taming the Underlying Challenges of Reliable Multihop Routing in Sensor Networks," in *SenSys 2003*, November 2003.
- [35] J. Liu, J. Reich, and F. Zhao, "Collaborative In-Network Processing for Target Tracking," *J. on Applied Signal Processing*, March 2003.
- [36] A. Arora, P. Dutta, and S. B. et al., "A Wireless Sensor Network for Target Detection, Classification, and Tracking," *Computer Networks (Elsevier) Systems*, 2003.
- [37] G. Simon and et. al., "Sensor Network-Based Countersniper System," in *SenSys 2004*, November 2004.
- [38] T. He, S. Krishnamurthy, J. A. Stankovic, and T. Abdelzaher, "An Energy-Efficient Surveillance System Using Wireless Sensor Networks," in *MobiSys'04*, June 2004.
- [39] R. Stoleru, T. He, and J. A. Stankovic, "Walking GPS: A Practical Solution for Localization in Manually Deployed Wireless Sensor Networks," in *EmNetS-I*, October 2004.
- [40] A. Okabe, B. Boots, K. Sugihara, and S. N. Chiu, *Spatial Tessellations: Concepts and Applications of Voronoi Diagrams*. Wiley, 2000.
- [41] G. Zhou, T. He, and J. A. Stankovic, "Impact of Radio Irregularity on Wireless Sensor Networks," in *MobiSys'04*, June 2004.
- [42] R. Williams, *Circle Coverings*. New York: Dover, 1979.
- [43] T. He, C. Huang, B. M. Blum, J. A. Stankovic, and T. Abdelzaher, "Range-Free Localization Schemes in Large-Scale Sensor Networks," in *MOBICOM'03*, September 2003.
- [44] T. He, B. M. Blum, J. A. Stankovic, and T. F. Abdelzaher, "AIDA: Adaptive Application Independent Data Aggregation in Wireless Sensor Networks," *ACM Transactions on Embedded Computing System, Special issue on Dynamically Adaptable Embedded Systems*, 2004.
- [45] D. H. Goldberg, P. J. Andreas G. Andreou, P. O. Pouliquen, L. Riddle, and R. Rosasco, "A Wake-up Detector for an Acoustic Surveillance Sensor Network: Algorithm and VLSI Implementation," in *IPSN'04*, 2004.
- [46] T. F. Abdelzaher, B. M. Blum, Q. Cao, D. Evans, J. George, S. George, T. He, L. Luo, S. H. Son, R. Stoleru, J. A. Stankovic, and A. Wood, "EnviroTrack: An Environmental Programming Model for Tracking Applications in Distributed Sensor Networks," in *ICDCS'04*, March 2004.
- [47] *Energizer current batteries datasheets*, Energizer, available at <http://www.energizer.com>.
- [48] *Mica2 aa battery pack service life test*, CrossBow, available at <http://www.xbow.com/Support/>.