University of Nebraska - Lincoln
DigitalCommons@University of Nebraska - Lincoln

CSE Conference and Workshop Papers | Computer Science and Engineering, Department of

1-1-2011

Analysis of Event Detection Delay in Wireless Sensor Networks

Yunbo Wang
University of Nebraska at Lincoln, yunbow@cse.unl.edu

Mehmet C. Vuran
University of Nebraska at Lincoln, mcvuran@cse.unl.edu

Steve Goddard
University of Nebraska – Lincoln, goddard@cse.unl.edu

Follow this and additional works at: http://digitalcommons.unl.edu/cseconfwork

Part of the Computer Sciences Commons

Wang, Yunbo; Vuran, Mehmet C.; and Goddard, Steve, "Analysis of Event Detection Delay in Wireless Sensor Networks" (2011). *CSE Conference and Workshop Papers*. Paper 217.
http://digitalcommons.unl.edu/cseconfwork/217

This Article is brought to you for free and open access by the Computer Science and Engineering, Department of at DigitalCommons@University of Nebraska - Lincoln. It has been accepted for inclusion in CSE Conference and Workshop Papers by an authorized administrator of DigitalCommons@University of Nebraska - Lincoln.

This paper was presented as part of the main technical program at IEEE INFOCOM 2011

2011 Proceedings IEEE INFOCOM,
doi: 10.1109/INFCOM.2011.5934912

Analysis of Event Detection Delay in Wireless Sensor Networks

Yunbo Wang Mehmet C. Vuran Steve Goddard
Department of Computer Science and Engineering
University of Nebraska-Lincoln
Email: {yunbow, mcvuran, goddard}@cse.unl.edu

***Abstract*—Emerging applications of wireless sensor networks (WSNs) require real-time event detection to be provided by the network. In a typical event monitoring WSN, multiple reports are generated by several nodes when a physical event occurs, and are then forwarded through multi-hop communication to a sink that detects the event. To improve the event detection reliability, usually timely delivery of a certain number of packets is required. Traditional timing analysis of WSNs are, however, either focused on individual packets or traffic flows from individual nodes. In this paper, a spatio-temporal fluid model is developed to capture the delay characteristics of event detection in large-scale WSNs. More specifically, the distribution of delay in event detection from multiple reports is modeled. Accordingly, metrics such as mean delay and soft delay bounds are analyzed for different network parameters. Motivated by the fact that queue build up in WSNs with low-rate traffic is negligible, a lower-complexity model is also developed. Testbed experiments and simulations are used to validate the accuracy of both approaches. The resulting framework can be utilized to analyze the effects of network and protocol parameters on event detection delay to realize real-time operation in WSNs. To the best of our knowledge, this is the first approach that provides a transient analysis of event detection delay when multiple reports via multi-hop communication are needed.**

I. Introduction

Due to their ubiquitous and flexible nature, wireless sensor networks (WSNs) have been intensively investigated recently for data monitoring applications [2]. In such applications, numerous sensor nodes are deployed in the space, and operate collaboratively to monitor, report, and react to various physical events. When an event of interest occurs, it is detected by sensor nodes. Reports are then generated and forwarded to a sink via multi-hop communication. According to the received reports, the sink may detect the event and perform appropriate actions, e.g., inform the forest administration in case of a fire. For such systems, it is crucial to have the reports delivered to the sink in a timely manner. Therefore, one of the most important performance metrics for event monitoring WSNs is the *event detection delay* [18], i.e., the delay between when an event occurs in the physical world and when sufficient number of packets are delivered to the sink. Clearly, the event detection delay consists of two parts: the *discovery delay* for individual nodes to sense and detect the event, and the *delivery delay* for the network to relay reports to the sink. Analyzing the event detection delay is a crucial task for real-time WSN applications, which require predictable event detection delay guarantees provided by the network.

Supported, in part, by grants from the National Science Foundation (0707975) and the Air Force Office of Scientific Research (FA9550-06-1-0375).

Compared to traditional wired and wireless networks, WSNs have two major distinctive characteristics, namely, low hardware profiles, and unpredictable environment conditions. These complicate the effort to analyze event detection delay in several aspects. First, the network is usually deployed with a high density due to low-cost nature of sensor nodes. Moreover, limited energy resources require low duty cycle operation. Thus, to maintain necessary amount of available nodes at any time, large amount of redundant nodes should be deployed. Second, it is often impractical to deploy sensor nodes at predefined locations. Instead, random deployment with a certain density is preferred.

The randomness in the network topology and the low power wireless communication results in a large variance in the end-to-end delay. Consequently, traditional *worse case delay bound* analysis [4, 13] has limited applicability in WSNs, as the worst case bounds are very loose and cannot accurately characterize the delay distribution. Instead, probabilistic models that can capture the distribution of communication delay are necessary [31].

Finally, as event reports from individual nodes can be unreliable, it is more desirable to detect event collectively from reports generated by multiple sensor nodes [1, 18]. Thus, event is generally considered to be detected only when a given number, n, of reports are received by the sink [17, 34].

To address these challenges, in this paper, an analytical framework is developed to capture the delay characteristics of event detection in large-scale WSNs. A spatio-temporal fluid model is developed to derive the distribution of event detection delay. Accordingly, the mean event detection delay and soft-delay bounds for event detection can be modeled. The soft-delay bound (or p-delay bound) for delay is defined as the delay within which an event is detected with a given probability p. Here, the given probability p is also called the *required reliability*. Indeed, given a reliability requirement, a lower p-delay bound indicates that the events can be reliably detected within a lower delay. Hence, the network is more desirable for real-time applications.

In the developed framework, the network topology is modeled by a spatial fluid network model [6, 7, 29]. Instead of individual nodes, the network is considered as a continuous fluid entity distributed on the network area. As a result, the complexity introduced by the random topology is greatly reduced. In addition, the traffic in the network is modeled based on a temporal fluid model [22, 25], which considers traffic generated by nodes as a continuous fluid flowing towards the destination. The resulting *spatio-temporal* fluid model captures

the dynamics of the generated packet flows. The empirical validations and simulation studies reveal that the developed model is suitable for high density networks and low traffic rate applications, common features of a large class of WSN applications.

Motivated by the fact that queue build up in low-rate traffic is negligible, a lower-complexity model is also developed. This model extends the end-to-end delay analysis for single packets in our previous work [31], and derives the event detection delay by first obtaining the end-to-end delay for each packet. This approach requires lower computational power.

To the best of our knowledge, this is the first work that investigates the event detection delay in large-scale multi-hop WSNs. Extensive testbed and simulation experiments validate both approaches in several network scenarios. For the scenarios in which the framework does not yield accurate results, potential reasons are briefly discussed. The resulting framework can be utilized to analyze the effects of network and protocol parameters on event detection delay to realize real-time operation in WSNs.

The rest of the paper is organized as follow: In Section II, a brief survey on the related work is presented. Section III provides the system model and formal definition of the problem. Then, in Section IV and Section V, the two approaches to characterize the event detection delay are described. The testbed and simulation evaluations are discussed in Section VI. Finally, the paper is concluded in Section VII.

II. Related Work

Characterizing timing performance in WSNs has been investigated in different contexts. For single packet multi-hop communication, the end-to-end delay distribution has been analyzed in our previous work in [31]. For the traffic flow, several models have been developed to analyze probabilistic *bounds* on delay. As an example, the concept of Network Calculus [8] is extended to derive probabilistic bounds for delay through worst case analysis [4, 13]. However, due to the randomness in and the low power nature of the communication links in WSNs, these worst case bounds cannot capture the stochastic characteristics of end-to-end delay. The communication capacity bounds for wireless networks or WSNs without duty cycle operation are investigated in [11, 14, 19, 26, 33]. However, the applicability of these models to WSNs is limited since in WSNs, the wireless channel utilization is often well below the transmission capacity as nodes are constantly forced into a sleeping state to preserve energy.

The existing studies on event detection delay in WSNs are either focused on (1) the event discovery delay, i.e., the delay until the event is detected by an individual node, or (2) the delivery delay in a broadcast network. In [5], assuming a uniform node deployment and a duty cycle based sensing scheme, an analytical model is developed to derive the distribution of the delay until a stationary or mobile physical event is discovered by any node in the network. In [9], events are considered as detected when it is discovered by a node *connected to the sink*. On the other hand, the communication delay for event detection is investigated in [16] for WSNs deployed in a star topology. When an event occurs, multiple sensors in the network discover it immediately, and transmit their report packets to the central controller. The probability distribution of the delivery delay for the first $n(n > 0)$ report packets is obtained using a hybrid automata model. However, this model cannot be easily employed for large-scale and multi-hop WSNs, where the model becomes intractable. On the contrary, we emphasize on the delay before the event is *detected by the sink*, which includes the event discovery delay and the event delivery delay. Moreover, by utilizing fluid-based models, large-scale multi-hop WSNs can be captured.

Fluid-based models have been widely exploited in IP network analysis [22, 25], and have recently been utilized in the analysis of WSNs [12]. Motivated by the fact that the individual packet behavior is insignificant when a flow is concerned, the traffic is considered as a continuous flow instead of individual packets. Accordingly, the complexity of the model can be greatly reduced. Furthermore, spatial fluid-based models have also been utilized recently in [7, 29] to model stationary properties, such as traffic rate and energy consumption for large-scale WSNs. These models greatly reduce the complexity of the (otherwise intractable) problem in either temporal or spatial domain. In our analytical framework, we develop a *spatio-temporal fluid model* for the analysis of event detection delay.

III. System Model and Problem Definitions

In this section, we first present the system model, including the random network topology model and a description for the network protocol in consideration. Then, the formal definitions of the problems are given.

A. Network Topology

In a network deployed to monitor a physical event, nodes are considered to be randomly located according to a Poisson point process, where the node density is ρ. A sink node is deployed at location $\boldsymbol{x}_0 = (x_0, y_0)$, as shown in Fig. 1.

Assume that at time $t = t_0$, a physical event occurs at location $\boldsymbol{x}_e = (x_e, y_e)$, which is called the event center, and lasts for duration T_e. As shown in Fig. 1, all sensor nodes within the detection range, r_e, can discover the event. Each sensor node periodically measures the physical world every t_e seconds using its attached sensors. During the event duration $[t_0, t_0 + T_e)$, whenever the value of the measurement satisfies a predefined rule, e.g., temperature higher than a given threshold, a report packet of size L is generated and is forwarded to the sink. Each sensor node is assumed to have the same sampling rate, but with a random phase shift, i.e., samples are taken unsynchronized among nodes. Therefore, there is a *discovery delay* between when the event occurs and when

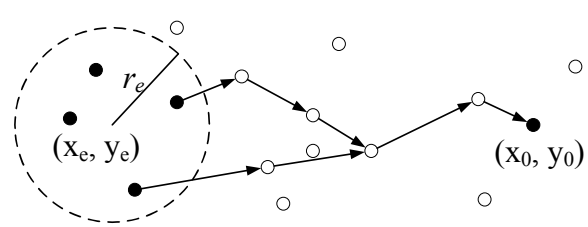


Fig. 1. The network including the sink and the event generation area.

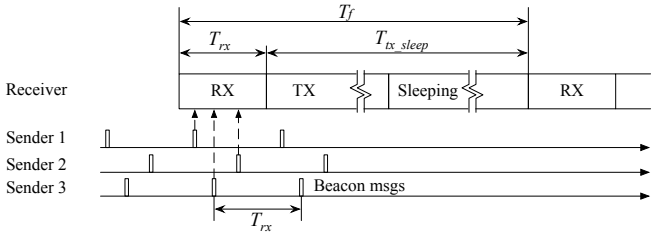


Fig. 2. The timing of node operations for the anycast protocol.

it is captured by individual nodes. Due to inherent noise in the sensor readings, $n(n \geq 1)$ readings from multiple sensor nodes are required at the sink to successfully detect the event occurrence. Accordingly, we define the following:

Definition 1. *An event is* **n-detected** *if n report packets for that event are received by the sink.*

Moreover, each node is implemented with a packet queue of maximum size, M.

B. MAC and Routing Protocol

The delay characteristics of a WSN is dependent on the MAC and routing protocols used in the network. To save communication energy, recent research has been focused on MAC protocols with duty cycle operations [3, 28]. In such protocols, nodes periodically enter active and sleeping states, and consume significantly less energy compared to nodes with MAC protocols that require the nodes to be always active. As a result of constantly entering sleeping state, the communication delay is often increased. To counter this drawback, opportunistic routing techniques, particularly anycast protocols, are often utilized along with a high node density to exploit node deployment redundancy [20, 21, 24, 27, 30]. If a node has packets to send, it first broadcasts a series of beacon messages. Then, one of responding neighbors is chosen as the next-hop node according to predefined rules (e.g., the first node that responds, or the closest node to the destination). Finally, the sender forwards the data packet to the chosen neighbor. While there is no single dominantly used anycast protocol in WSNs, in this paper, we assume that the following representative protocol is used.

The anycast protocol operation is depicted in Fig. 2. Each node, except the sink, operates in a duty cycle, and the length of the cycle (frame length) is T_f. Each cycle is divided into two phases. During the first phase, the listening phase, nodes listen on the channel for any possible incoming traffic. The second phase is the transmission and sleeping phase, in which nodes first try to transmit every packet in the queue. After all packets are transmitted, they turn off radio transceivers to save energy. The duration of these two phases are denoted as T_{rx} and T_{tx_sleep}, respectively. The duty cycle, ξ, is defined as the proportion of the listening phase duration in a cycle, i.e., $\xi = T_{rx}/T_f$. To obtain a long network lifetime, it is desirable to have a very low duty cycle and hence, generally, $T_{rx} << T_f$. Nodes are assumed *not* to be synchronized.

The packet transmission process is as follows: When a node has a packet to send, it first broadcasts short beacon messages periodically. If any other node is listening on the channel and receives the beacon message, it checks the following criteria: (1) it is closer to the sink than the sender; (2) the beacon message is received with a higher signal to noise ratio (SNR), ψ, than a given threshold, ψ^{th}. The second criterion is to ensure a low packet error rate due to channel quality. If both of the criteria are satisfied, the receiver sends back a CTS message. There may be multiple nodes that receive the beacon message. Thus, the sender may receive multiple CTS messages. The first node that responds with a CTS message is chosen as the next-hop node. Finally, the sender transmits the data packet to it.

The transmission interval of beacon messages is set equal to T_{rx}, as shown in Fig. 2, to ensure that other nodes can receive the beacon messages when they are listening. Therefore, during the listening phase, each node receives a beacon message from its neighbors, if they transmit beacon messages. We assume that all messages from different nodes do not collide with each other. This is a valid assumption, because the duty cycle is usually very small in monitoring applications. Moreover, the beacon and CTS messages are very short and are unlikely to collide. Although data packets may be longer, their length is still usually very small compared to the listening period. In the rare event where data packets do collide, their senders can utilize retransmissions after a short amount of delay to ensure delivery. For example, in a typical monitoring WSN application, the operation cycle T_f may be set to 10 s, and the listening phase duration T_{rx} may be set to 100ms to achieve a 1% duty cycle, as shown in Fig. 2. The transmission duration of a beacon message or a CTS message is usually less than 1ms, and the transmission duration for a data packet with 40 bytes is less than 2ms for many WSN platforms such as MicaZ and TelosB. The collision probability in this case is minimal and can be neglected. The testbed evaluations reveal that these assumptions are reasonable as discussed in Section VI.

Note that during a listening phase, a node can only receive a single beacon message from any other node. Thus, at most one packet can be transmitted from node a to node b for any neighbor nodes (a, b) during a duty cycle T_f. However, a node may receive multiple beacon messages and respond to them during a listening phase. Thus, it is possible for a node to receive multiple data packets in each listening phase from multiple senders. In such scenario, all packets received are stored in the queue. New packets are dropped if they arrive when the queue is full.

Packets are transmitted in a FIFO basis, until buffered packets are all transmitted. A node may transmit multiple packets to multiple neighbors, but can only transmit one packet to each single neighbor. After all packets are transmitted, the node turns off its transceiver to save energy, until the next listening phase starts. If at the end of the transmission/sleeping phase, there are still packets not transmitted, the node stops broadcasting beacon messages and begins listening. The beacon message for the current packet will be resumed in the next transmission/sleeping phase.

C. Problem Definitions

As explained in Section I, several random factors in the topology and node operation affect the communication in the

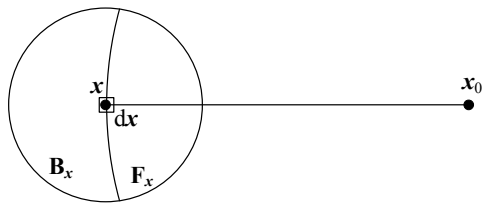


Fig. 3. Feasible region $\mathbb{F}_{\boldsymbol{x}}$ and backward region $\mathbb{B}_{\boldsymbol{x}}$ around location $\boldsymbol{x}$.

network. Accordingly, the paths from detecting sensors to the sink are dynamically generated and can be considered random. Accordingly, the delay characteristics of event detection is modeled based on the following definitions.

Definition 2. *The* **n-delay** *of an event is the delay between when the physical event occurs and when the event is n-detected.*

Definition 3. *The* **(p, n)-delay bound** *of an event is delay within which the event is n-detected with probability p.*

It is assumed that no in-network processing, such as aggregation, repacking, etc., is utilized in the network and their effects are left as a future work. To evaluate the delay characteristics of event detection in WSNs, given network and protocol parameters, n, and p; we are interested in the following problems:

- What is the n-delay distribution of an event?
- What is the average n-delay of an event?
- What is the (p, n)-delay bound of an event?

In Section IV and Section V, the proposed spatio-temporal fluid models are presented to address these questions.

IV. Transient Analysis of Event Detection

In this section, the spatio-temporal fluid model is presented. The network is represented by a continuous fluid entity distributed in the entire network area. Furthermore, the traffic is not considered as individual packets, but a continuous packet fluid. By utilizing a spatio-temporal fluid model, the complexity of the problem in both spatial and temporal domains is reduced, and becomes tractable. The testbed and simulation evaluations (Section VI) reveal that the fluid approximation accurately models the delay characteristics.

Consider a location in the network area denoted as $\boldsymbol{x} = (x, y)$. Since the fluid network model regards the nodes as a fluid entity over the entire space, in an infinitesimal area around location $\boldsymbol{x}$ with size $\mathrm{d}\boldsymbol{x}$ [1], the amount of nodes is $\rho \mathrm{d}\boldsymbol{x}$, where ρ is the node density. We also denote the *feasible region* of $\boldsymbol{x}$ (the region in the transmission range of $\boldsymbol{x}$ and is closer to the sink) as $\mathbb{F}_{\boldsymbol{x}}$, and the *backward region* of $\boldsymbol{x}$ (the region in the transmission range of $\boldsymbol{x}$ and is farther to the sink) as $\mathbb{B}_{\boldsymbol{x}}$, as shown in Fig. 3.

To describe the fluid traffic in the spatial fluid network, the following traffic concepts are introduced:

Definition 4. *The generated, incoming, and outgoing traffic rate density for an infinitesimal area $\mathrm{d}\boldsymbol{x}$ is respectively defined as the average number of packets generated, received, and transmitted by the nodes within the area, if any, in an infinitesimal duration $\mathrm{d}t$, divided by the duration $\mathrm{d}t$, and the size of the area $\mathrm{d}\boldsymbol{x}$.*

[1]With a slight abuse of denotation, this infinitesimal area is henceforth denoted as $\mathrm{d}\boldsymbol{x}$.

In other words, the traffic rate densities define the speed at which packets are generated, received, and transmitted in unit space, respectively. In the transient analysis, their values change over time, and thus, are functions of t. The generated, incoming, and outgoing traffic rate density are denoted as $g_{\boldsymbol{x}}(t)$, $\lambda_{\boldsymbol{x}}(t)$, and $\omega_{\boldsymbol{x}}(t)$, respectively.

Definition 5. *The buffered traffic density for an infinitesimal area $\mathrm{d}\boldsymbol{x}$ is defined as the average number of packets buffered in the queue by the nodes within the area divided by the size of the area $\mathrm{d}\boldsymbol{x}$.*

The buffered traffic density is also a function of t, and is denoted as $q_{\boldsymbol{x}}(t)$.

In the following, we derive the set of equations that describe the fluid traffic characteristics of the network after $t = t_0$. Without loss of generality, let $t_0 = 0$. For each node, the generated traffic rate density is given by

$$g_{\boldsymbol{x}}(t) = \begin{cases} \dfrac{\rho}{t_e}, & |\boldsymbol{x} - \boldsymbol{x}_e| < r_e, \text{and } 0 \le t < T_e, \\ 0, & \text{otherwise}, \end{cases} \tag{1}$$

where ρ is the density, t_e is the reporting interval, and $|\boldsymbol{x} - \boldsymbol{x}_e|$ denotes the Euclidean distance between $\boldsymbol{x}$ and $\boldsymbol{x}_e$. During an infinitesimal duration $\mathrm{d}t$, the amount of arriving traffic, along with the traffic already stored in the queue is

$$a_{\boldsymbol{x}}(t) = q_{\boldsymbol{x}}(t) + (\lambda_{\boldsymbol{x}}(t) + g_{\boldsymbol{x}}(t)) \cdot \mathrm{d}t. \tag{2}$$

which is the available traffic that needs to be transmitted.

For each infinitesimal area $\mathrm{d}\boldsymbol{x}'$ in the feasible forwarding region $\mathbb{F}_{\boldsymbol{x}}$ of $\boldsymbol{x}$, the amount of nodes with good channel quality is

$$c_{\boldsymbol{x},\boldsymbol{x}'} = \rho \cdot p_{\boldsymbol{x},\boldsymbol{x}'}(\psi^{Th}), \tag{3}$$

where $p_{\boldsymbol{x},\boldsymbol{x}'}(\psi^{Th})$ is the probability that the CTS message sent from a node at $\boldsymbol{x}'$ has a higher SNR than a given threshold ψ^{th} when received by the node at $\boldsymbol{x}$ ((10) in [35]). Thus, the total amount of nodes in $\mathbb{F}_{\boldsymbol{x}}$ with good channel quality is

$$c_{\mathbb{F}_{\boldsymbol{x}}} = \int_{\mathbb{F}_{\boldsymbol{x}}} c_{\boldsymbol{x},\boldsymbol{x}'} \mathrm{d}\boldsymbol{x}'. \tag{4}$$

Note that between any pair of nodes, at most one packet can be transmitted in a cycle T_f. Thus the maximum amount of traffic transmitted during a cycle T_f from $\mathrm{d}\boldsymbol{x}$ to anywhere in $\mathbb{F}_{\boldsymbol{x}}$ is

$$\Omega^m_{\boldsymbol{x}} = \rho \mathrm{d}\boldsymbol{x} \cdot c_{\mathbb{F}_{\boldsymbol{x}}}. \tag{5}$$

Since the traffic is considered as a fluid and a packet takes one cycle to be transmitted between a pair of nodes, in $\mathrm{d}t$, the maximum amount of traffic sent from $\mathrm{d}\boldsymbol{x}$ is $\Omega^m_{\boldsymbol{x}} \cdot \mathrm{d}t / T_f$. In the case where each node in $\mathrm{d}\boldsymbol{x}$ has less than 1 available packet in its queue, i.e., $a_{\boldsymbol{x}}(t) < 1 \cdot \rho$, it still takes an entire cycle to transmit them. In this case the actual transmitted traffic during

$\mathrm{d}t$ is $\frac{a_{\boldsymbol{x}}(t)}{1\cdot\rho}\cdot\Omega_{\boldsymbol{x}}^{m}\cdot\frac{\mathrm{d}t}{T_f}$. Accordingly, the transmitted traffic rate density at $\boldsymbol{x}$ is

$$\omega_{\boldsymbol{x}}(t)=\min\left[1,\frac{a_{\boldsymbol{x}}(t)}{1\cdot\rho}\right]\Omega_{\boldsymbol{x}}^{m}\frac{\mathrm{d}t}{T_f}\cdot\frac{1}{\mathrm{d}\boldsymbol{x}\mathrm{d}t}=\min\left[a_{\boldsymbol{x}}(t),\rho\right]\frac{c_{\mathbb{F}_{\boldsymbol{x}}}}{T_f}, \tag{6}$$

where $a_{\boldsymbol{x}}(t)$ is the available traffic density given by (2).

The outgoing traffic in each infinitesimal area is equally distributed into every nodes with good channel quality in its feasible region. Thus, the incoming traffic rate density, $\lambda_{\boldsymbol{x}}(t)$, that is received from each infinitesimal area in the backward region, $\mathbb{B}_{\boldsymbol{x}}$, is given by

$$\lambda_{\boldsymbol{x}}(t)=\int_{\mathbb{B}_{\boldsymbol{x}}}\omega_{\boldsymbol{x}'}(t)\cdot\frac{c_{\boldsymbol{x}',\boldsymbol{x}}}{c_{\mathbb{F}_{\boldsymbol{x}'}}}\mathrm{d}\boldsymbol{x}'. \tag{7}$$

Within duration $\mathrm{d}t$, the change in buffered traffic density is

$$\mathrm{d}q_{\boldsymbol{x}}(t)=(g_{\boldsymbol{x}}(t)+\lambda_{\boldsymbol{x}}(t)-\omega_{\boldsymbol{x}}(t))\mathrm{d}t, \tag{8}$$

and the buffered traffic density at time $t+\mathrm{d}t$ changes to

$$q_{\boldsymbol{x}}(t+\mathrm{d}t)=q_{\boldsymbol{x}}(t)+\mathrm{d}q_{\boldsymbol{x}}(t) \tag{9}$$

Thus, (6), (7), (8) and (9) describe the traffic dynamics of the network after $t=t_0$. Given the initial value of $q_{\boldsymbol{x}}(t_0)$, the traffic rates in the network can be evaluated for any time instance $t>t_0$. Accordingly, the total incoming traffic rate at the sink, which models the total number of packets received by the sink, can be obtained. Note that within the transmission range of the sink, the outgoing traffic rate density in (6) becomes

$$\omega_{\boldsymbol{x}}(t)=a_{\boldsymbol{x}}(t), \tag{10}$$

since the sink is always awake and the traffic can all be transmitted to the sink directly. Moreover, for these nodes, in (7), the backward region $\mathbb{B}_{\boldsymbol{x}}$ excludes the areas within the transmission range of the sink. Then, at the sink, the incoming traffic rate is calculated as

$$\Lambda(t)=\int_{\boldsymbol{x}:|\boldsymbol{x}-\boldsymbol{x}_0|\le R}\omega_{\boldsymbol{x}}(t)\mathrm{d}\boldsymbol{x}. \tag{11}$$

To calculate the incoming traffic rate at the sink, the entire network area is descretized into small areas, and the time is divided into small time steps. Initially, the buffered traffic density for every infinitesimal area in the network at time $t=0$ is $q_{\boldsymbol{x}}^0$. $\lambda_{\boldsymbol{x}}(t)$ and $\omega_{\boldsymbol{x}}(t)$ are set as 0. Then, $\omega_{\boldsymbol{x}}(t)$ and $\lambda_{\boldsymbol{x}}(t)$ are calculated using (6) (7), respectively. Then, $q_{\boldsymbol{x}}(t)$ is updated for the next time step according to (8). This process is repeated for each time step, and $\Lambda(t)$ as a function of t is obtained.

Although the packets are generated with a periodic pattern, the randomness introduced by the routing path and the communication delays results in stochastic behavior for the arrival of packets after multiple hops at the sink, especially in large-scale networks. More specifically, empirical experiments reveal that the traffic arrival process can be approximated by a Poisson process [31]. To obtain the n-delay distribution from $\Lambda(t)$, the traffic arrival process is considered a Poisson process with variable rate according to $\Lambda(t)$. The evaluations in Section VI also validate the accuracy of this assumption. Consequently, the n-delay distribution, $f_n(t)$, of the nonhomogeneous Poisson process is given by [15, Ch. 2.4]:

$$f_n(t)=\frac{[\hat{\Lambda}(t)]^{(n-1)}\Lambda(t)\mathrm{e}^{-\hat{\Lambda}(t)}}{(n-1)!}, \tag{12}$$

where $\hat{\Lambda}(t)$ is the integral of $\Lambda(t)$ over duration $(0,t]$.

Finally, the average n-delay and the (p,n)-delay bound of an event are

$$\mu(n)=\int_0^{\infty}tf_n(t)\mathrm{d}t, \tag{13}$$

$$j(p,n)=f_n^{-1}(p), \tag{14}$$

respectively.

V. Simplified Delay Model

The spatio-temporal fluid model presented in Section IV greatly lowers the complexity of the problem. In the model, the entire network area is descretized into small areas, and the traffic rates are calculated for each small area in each time step. To achieve a high accuracy, the size of small areas and the duration of time steps are usually chosen very small. In this section, we provide a simplified model to further reduce the calculation complexity. In this simplified model, the network area is divided into small rings. Thus, the spatial calculation complexity is reduced from 2D to 1D.

The simplified model assumes that the traffic is very low in the network, which is typical for many WSN applications. Thus, the queueing effect can be neglected. Moreover, based on the channel-aware next-hop selection explained in Section III, it is assumed that the channel error is negligible within a transmission range of R. For a node located at $\boldsymbol{x}$, after it receives a packet (locally generated or forwarded), in the duration t, the probability that there is no node in its feasible forwarding region $\mathbb{F}_{\boldsymbol{x}}$ waking up is

$$\begin{aligned}p_{\boldsymbol{x}}^{nf}(t)&\approx\prod_{l=|\boldsymbol{x}|-R}^{|\boldsymbol{x}|}\prod_{\theta=-\theta_{max}}^{\theta_{max}}\left[1-\rho\mathrm{d}l\mathrm{d}\theta p^{ol}(t)\right]\\&=\left[1-\rho\mathrm{d}l\mathrm{d}\theta p^{ol}(t)\right]^{\frac{A_{\mathbb{F}_{\boldsymbol{x}}}}{\mathrm{d}l\mathrm{d}\theta}}=e^{-A_{\mathbb{F}_{\boldsymbol{x}}}\rho p^{ol}(t)},\end{aligned} \tag{15}$$

where the product is conducted over $A_{\mathbb{F}_{\boldsymbol{x}}}$, divided according to the polar coordinates originated at $\boldsymbol{x}$, ρ is the network density, $A_{\mathbb{F}_{\boldsymbol{x}}}$ is the size of $\mathbb{F}_{\boldsymbol{x}}$, and $p^{ol}(t)$ is the probability that a node in each region wakes up during the period t. Also, $\theta_{max}=(|\boldsymbol{x}|^2+|\boldsymbol{x}'|^2-R^2)/(2|\boldsymbol{x}|\cdot|\boldsymbol{x}'|)$, where $\boldsymbol{x}'$ is the location (l,θ). Since each node wakes up at uniformly distributed times, we have

$$p^{ol}(t)=\begin{cases}\dfrac{t}{T_f}, & 0\le t\le T_f\\ 1, & t>T_f\end{cases}. \tag{16}$$

Therefore, the probability that at least one node in $\mathbb{F}_{\boldsymbol{x}}$ wakes up during t is

$$p_{\boldsymbol{x}}^{f}(t)=1-p_{\boldsymbol{x}}^{nf}(t)=\begin{cases}1-e^{-A_{\mathbb{F}_{\boldsymbol{x}}}\rho t/T_f}, & 0\le t\le T_f\\ 1-e^{-A_{\mathbb{F}_{\boldsymbol{x}}}\rho}, & t>T_f\end{cases}. \tag{17}$$

This is exactly the cdf of the single hop delay. Therefore, the pdf of single hop delay for a node at $\boldsymbol{x}$ is

$$f^s_{\boldsymbol{x}}(t) = \mathrm{d}p^f_{\boldsymbol{x}}(t)/\mathrm{d}t = \begin{cases} \frac{A_{\mathbb{F}_{\boldsymbol{x}}}\rho}{T_f} e^{-A_{\mathbb{F}_{\boldsymbol{x}}}\rho t/T_f}, & 0 \le t \le T_f \\ 0, & t > T_f \end{cases}. \tag{18}$$

The end-to-end delay distribution from location $\boldsymbol{x}$ to the sink can be found as the convolution of single-hop delay distributions in the path [31, 32]. Thus, the pdf of end-to-end delay from $\boldsymbol{x}$ to the sink is

$$f^{e2e}_{\boldsymbol{x}}(t) = \sum_{l=|\boldsymbol{x}|-R}^{|\boldsymbol{x}|} \sum_{\theta=-\theta_{max}}^{\theta_{max}} f^{e2e}_{\boldsymbol{x}'} * f^s_{\boldsymbol{x}}(t), \tag{19}$$

where $\boldsymbol{x}'$ is the location (l, θ). Note that since the queueing effect is neglected, the nodes with the same distance to the sink have the same end-to-end delay to the sink. Therefore, the end-to-end delay distribution is calculated only once for all nodes with the same distance to the sink. This fact results in a significant reduction on the calculation time.

Suppose the packet generation function for a node at $\boldsymbol{x}$ is $g_{\boldsymbol{x}}(t)$, then the packet reception rate from $\boldsymbol{x}$ by the sink is

$$\lambda_{\boldsymbol{x}}(t) = g_{\boldsymbol{x}} * f^{e2e}_{\boldsymbol{x}}(t). \tag{20}$$

Then, the packet reception rate at the sink is the sum of traffic generated from each location in the event detection region. Thus,

$$\Lambda(t) = \sum_{l=|\boldsymbol{x}_e|-r_e}^{|\boldsymbol{x}_e|+r_e} \sum_{\beta=-\theta^e_{max}}^{\theta^e_{max}} g_{\boldsymbol{x}} * f^{e2e}_{\boldsymbol{x}}(t), \tag{21}$$

where $\boldsymbol{x}_e$ is the location of the event center, r_e is the event detection range, and $\theta^e_{max} = (|\boldsymbol{x}_e|^2 + |\boldsymbol{x}|^2 - r_e^2)/(2|\boldsymbol{x}_e| \cdot |\boldsymbol{x}|)$.

Finally, the average n-delay and the (p, n)-delay bound of an event are obtained by using (21) in (12) - (14).

VI. Testbed Validation and Simulation Results

To evaluate the accuracy of our proposed analytical framework, testbed experiments and simulations are conducted. The average n-delay and the (p, n)-delay bound of an event in the experiments and simulations are used to compare against the framework. The spatio-temporal fluid model in the framework is implemented using C++ and the simplified model is implemented using MATLAB. The computing environment is a computer with a Xeon 5150 CPU working at 2.66GHz and 2GB RAM. In this section, we show that our models provide a high accuracy against both empirical experiments and simulations.

A. Validation of the Event Detection Delay Analysis

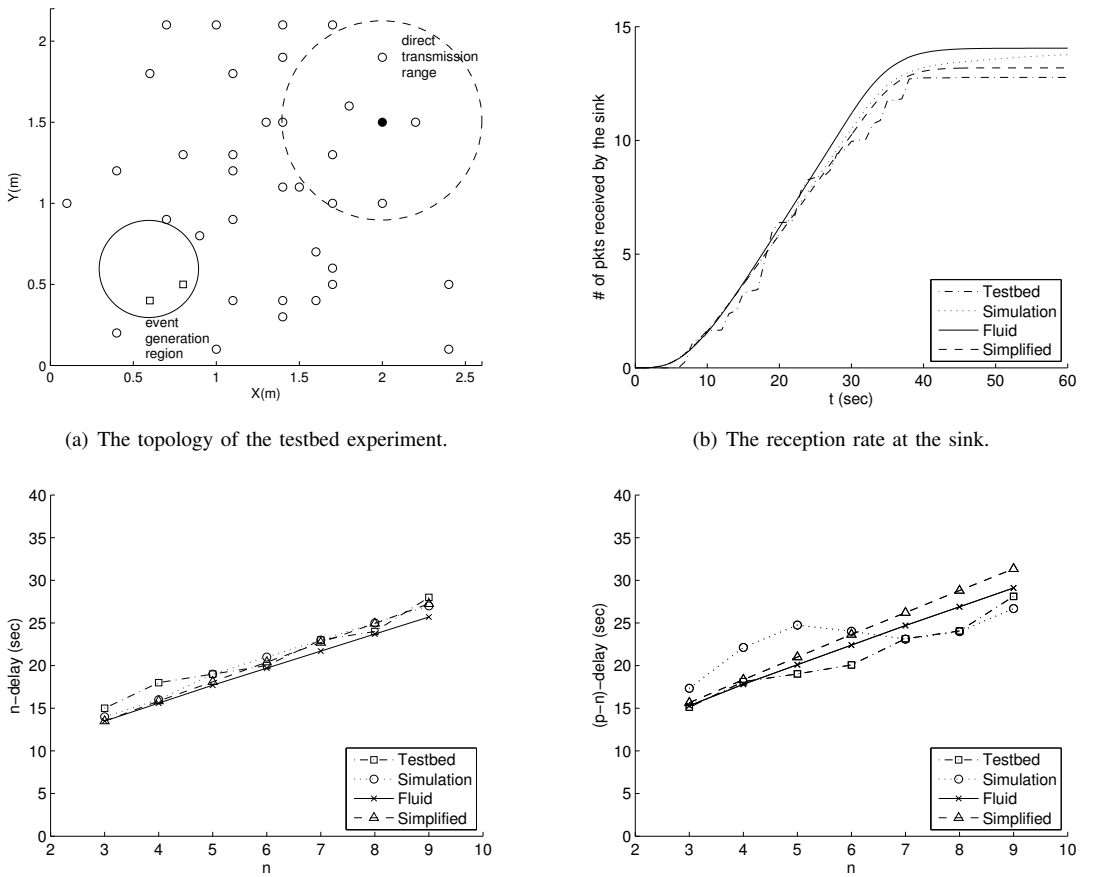
We first present the results of the testbed experiments. Our testbed consists of 40 Crossbow TelosB motes. The nodes are randomly placed in a rectangular area of size $2 \times 2.4\text{m}^2$, as shown in Fig. 4(a). The node density is thus 7.6/m^2. The sink is located at $(1.5\text{m}, 1.9\text{m})$ and is marked by a solid dot in Fig. 4(a). The data packet size is $l_p = 40$ bytes, whereas the beacon message and the CTS response message have the same size of $l_m = 22$ bytes. The cycle of each node is $T_f = 5$s, during which the wake period is $T_{rx} = 0.1$s, thus the duty cycle is $\xi = 0.02$. The beacon transmission duration is $T_{tx} = 1.6$ ms and the beacon transmission interval is 0.1s. The transmission power is set to -15dBm for all nodes. The threshold radius r^{th} is set to 0.6m, within which all nodes only transmit packets to the sink. The SNR threshold is set to $\psi^{th} = 10$dB. The event center is located at $(0.5\text{m}, 0.5\text{m})$. Each node in the network boots up at random time instances. Therefore, they are not synchronized. At time t_0, each of the nodes within r_e of the event center (marked as squares in Fig. 4(a)) starts to generate a series of packets with interval 4s, and then the generated packets are forwarded to the sink. After 30s, the nodes stop to generate packets. The experiment is conducted for 62 times. In each test, the delay for all packets received by the sink is logged, and the average total number of packets received as a function of time t after t_0 is depicted in Fig. 4(b).

The experiment is also conducted using TOSSIM [23] with the same parameters. Instead of a fixed topology, 100 randomly generated network topologies with the same area and density are used and for each random topology, the simulation is conducted for 5 trials. Then, the total number of packets received at the sink is calculated over the trials. The results are shown in Fig. 4(b) along with the results given by the two analytical models in (11) and (21). As can be seen in Fig. 4(b), both testbed experiments and simulations validate the models. To evaluate the accuracy of the models in terms of the average n-delay and (p, n)-delay bound of event detection, from the testbed and simulation result the mean delay for the first $n = 3$ to 9 packets are calculated respectively. The (p, n)-delay bound for $p = 0.75$ is also calculated for different n's. The results are shown in Fig. 4(c) and Fig. 4(d). Testbed and simulation results are within 5% of the model. For the majority of the cases, testbed and simulation results are within 5% of the model. Moreover, the results also confirm the accuracy of TOSSIM simulatons, which are used in the following for evaluations of the proposed framework in larger-scale networks.

B. Validation in Larger-Scale Networks

To further evaluate the accuracy of the analytical framework in larger-scale networks, extensive simulations are conducted. Network density, ρ, and the sensing interval, t_e, are varied to observe their impact on the event detection delay. Unless otherwise noted, the following parameters are used in the evaluations: The nodes are randomly generated in a square area of size $60 \times 60\text{m}^2$, according to a Poisson point process with density $\rho = 0.2$ nodes/m^2. The transmission power is 0dB, which corresponds to a transmission range of roughly 10m. The cycle length is $T_f = 10$s, in which the listening period is $T_{rx} = 0.1$s, corresponding to a duty cycle of $\xi = 0.01$. The packet sensing interval is $t_e = 4$s. The event detection range is $r_e = 5$m. The sink is located at $(0, 0)$m, and the event center is located at $(30, 30)$m. Thus, the distance between the event center and the sink is $d_{es} = 42.4$m. Other parameters are the same as those in Section VI-A.

In Figs. 5(a)-5(d), the effects of sensing interval are shown. In different simulations, the sensing interval t_e is set to 1s, 2s, 3s and 6s respectively for each node, corresponding to a packet generation rate of 1, 0.5, 0.333, and 0.167 pkt/s, respectively. The time instances of packet arrivals are logged at the sink,



(a) The topology of the testbed experiment.

(b) The reception rate at the sink.

(c) The mean event delay.

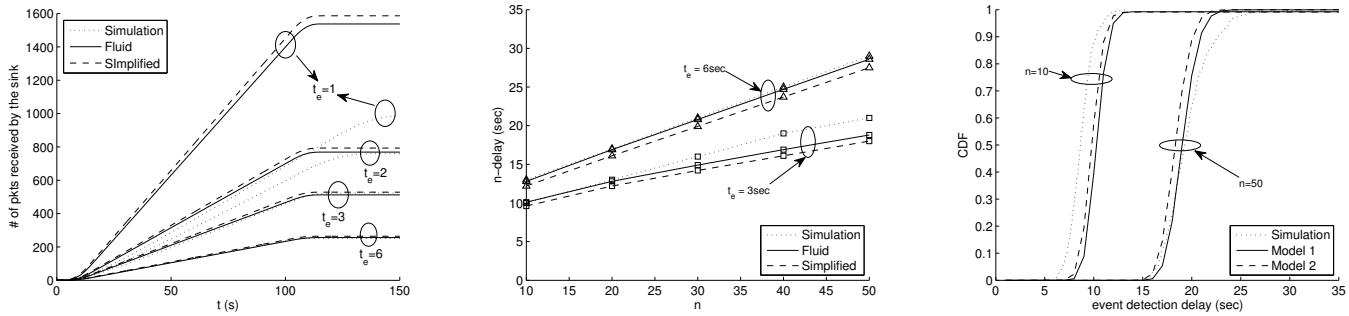
(d) The (p, n)-delay bound of the event delay for $p = 0.75$.

Fig. 4. The map of the testbed experiment, and the results of testbed experiment, the simulation and the models.

and the average total number of packets received over time is plotted in Fig. 5(a) along with the model results. It can be observed that for larger sensing intervals, i.e., 3s and 6s, both models are accurate with an error less than 5%, whereas for smaller sensing intervals, i.e., 1s and 2s, the models accuracy is lower. This is because with a lower sensing interval, the generated traffic rate is higher. As a result, the received traffic rate at the sink is affected by the clustering effect discussed in Section VI-D.

The average n-delay of an event for $n = 10, 20, 30, 40, 50$ packets are shown in Fig. 5(b) for $t_e = 2, 3$s respectively. Moreover, Fig. 5(c) depicts the cdf of event detection delay for $n = 10$ and 50 with sensing interval $t_e = 3$s. The average 10- and 50-delays are also shown in Fig. 5(d) with varying sensing rate (i.e., the inverse of the sensing interval). In can be shown that the delay reduces when the sensing rate increases, as expected. Note that as Fig. 5(b) suggests, for $n = 10$, the mean event detection delay for $t_e = 6$ only increases for about 2.5s from the mean delay for $t_e = 3$s. This is because for the first 10 packets, the majority of the event detection delay is caused by the packet communication to the sink, no matter how fast packets are generated. In practice, it may be a good idea to set a low sensing rate for sensing nodes if only a few packets are required to detect the event. This saves a great amount of sensing energy with a relatively small tradeoff of event detection delay.

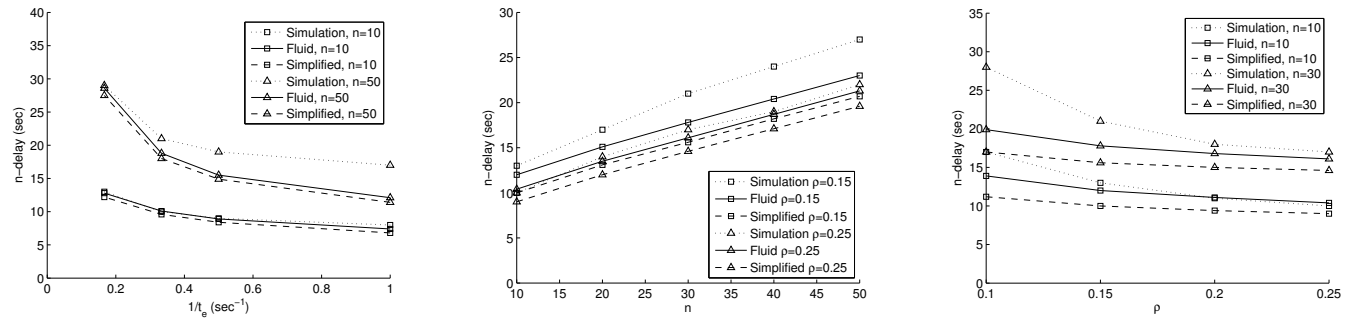
Next, the effects of network density are investigated for values of 0.1, 0.15, 0.2, and 0.25 nodes/m^2. Note that although the network density is changed, the total number of packets generated in the event area remains the same. This is achieved by setting the sensing interval t_e to 2, 3, 4, and 5s, respectively. By fixing the input traffic, the forwarding capacity of the network with changing density can be analyzed. The mean event detection delay for each density is shown in Fig. 5(e) as a function of the number of packets n required to detect the event. The 10- and 50-delays of an event are shown in Fig. 5(f). The figures show that when the network density is lower, the event detection delay slightly increases. This is because when the density is high, each node in the network generally waits for less time before another node wakes up and becomes available to forward its traffic. Thus, the transmission delay is lower, and the event detection delay is also reduced. However, the traffic reception rate at the sink is still limited by the packet generation rate, which is the same for all cases. Therefore, the average n-delay and (p, n)-delay bound of event detection does not change too much when the density is changed. This suggests in practice, to save sensing energy, the sensing rate of sensor nodes can be reduced. To compensate for the increased event detection delay, additional nodes can be deployed to increase the density.



(a) packet reception over time for various sensing intervals t_e.

(b) Mean delay vs. n for various sensing intervals t_e.

(c) Delay distribution for various sensing intervals t_e.



(d) Mean delay vs. sensing interval for various n.

(e) Mean delay vs. n for various node densities ρ.

(f) Mean delay vs. node density for various n.

Fig. 5. Mean delay and delay distribution for larger-scale networks. Analytical results are accurate compared to simulations in most cases.

Note that when the density is as low as 0.15 nodes/m^2, the average n-delay and (p, n)-delay bound from the simulation are higher than the model predictions. This is because when the network density is high, nodes operate well below the forwarding capability. On the other hand, when the density is lower, the network supports less amount of traffic forwarded to the sink. Thus, the detection delay is limited by the lower capability. The error of model prediction is due to the clustering effect discussed in Section VI-D.

C. Comparison Between the Models

In this section, we briefly present the difference between the two proposed models. By assuming that the nodes with the same distance to the sink have the same end-to-end delay to the sink, the simplified model requires $\mathrm{O}(\sqrt{A})$ time, where A is the area of the network. On the other hand, since the spatio-temporal fluid model calculates the traffic rates for each location in the entire 2D network, it requires $\mathrm{O}(A)$ time. The simplified model significantly outperforms the fluid model in terms of calculation efficiency. Note that both models yield the result in a significantly less time than simulations. For a typical network of 400 nodes, the simulation takes more than one day to complete, while the complete fluid model takes around 10 minutes to calculate, and the simplified only takes less than 1 minute.

On the other hand, the simplified model becomes inaccurate when the nodes with the same distance to the sink do not have the same end-to-end delay. An example is a non-regular network where nodes density varies over the space. The complete fluid model, however, can be extended to provide accurate result in such networks when the density ρ in (1) - (6) by $\rho(\boldsymbol{x})$, a density function of corresponding location $\boldsymbol{x}$.

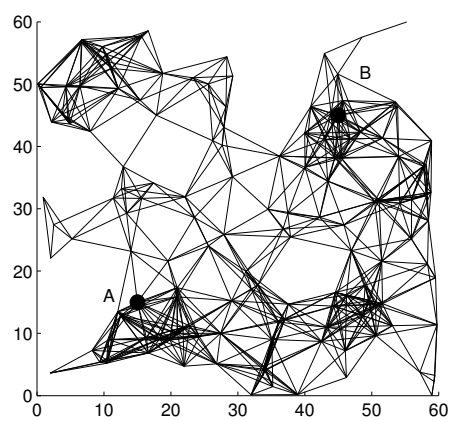


Fig. 6. Bottlenecks in random WSNs

D. Limitations of the Models

It is shown in Figs. 4 and 5 that both proposed models yield accurate results. The only cases where the result is less accurate is when the density is low, or when the traffic rate is high. Although such scenarios are generally not typical in WSN applications, it is important to point out that this is because by considering the nodes as a uniformly distributed fluid over the space, the bottlenecks of random network is neglected. In Fig. 6 an intuitive view is provided on how the bottlenecks are formed in a random network. As can be seen in Fig. 6, the nodes form

multiple clusters within which the nodes have a high degree of connection and less degree among clusters. This suggests that the transport capacity between A and B, which is usually dependent on the minimum cutset between the two nodes, is limited by the few paths across the clusters. These paths form the bottlenecks of communication [10]. Since the spatio-temporal fluid model assumes a uniform node distribution, it does not capture the bottlenecks, which may cause a higher detection delay when the network density is low, or when the traffic rate is high.

Note that although communication capacity bounds for wireless network communications without duty cycle operations are investigated in [11, 14, 19, 26], as far as we know, the exact solution for communication capacity in random WSNs with duty cycle operations is still an open research issue.

VII. Conclusions

In this paper, an analytical framework is proposed to model the event detection in WSNs. In the framework, a spatio-temporal fluid model is utilized to obtain the distribution of the event detection delay. The average delay and soft delay bounds are then obtained. To reduce the calculation complexity, a simplified model is also proposed, motivated by the fact that the queue build up in WSNs is negligible. Testbed experiments and simulations are used to validate the accuracy of both approaches. The resulting framework can be utilized to analyze the effects of network and protocol parameters on event detection delay to realize real-time operation in WSNs. To the best of our knowledge, this is the first approach that provides a transient analysis of event detection delay when multiple reports via multi-hop communication are needed.

References

[1] Ö. B. Akan and I. Akyildiz, "Event-to-sink reliable transport in wireless sensor networks," *IEEE/ACM Trans. on Networking*, vol. 13, no. 5, pp. 1003–1016, Oct 2005.

[2] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, "Wireless sensor networks: a survey," *Computer Networks (Elsevier) Journal*, vol. 38, no. 4, pp. 393–422, Mar 2002.

[3] M. Buettner, G. V. Yee, E. Anderson, and R. Han, "X-MAC: a short preamble MAC protocol for duty-cycled wireless sensor networks," in *SenSys '06: Proceedings of the 4th international conference on Embedded networked sensor systems*, pp. 307–320, New York, NY, May 2006.

[4] A. Burchard, J. Liebeherr, and S. Patek, "A min-plus calculus for end-to-end statistical service guarantees," *Information Theory, IEEE Transactions on*, vol. 52, no. 9, pp. 4105–4114, Sep 2006.

[5] Q. Cao, T. Yan, J. Stankovic, and T. Abdelzaher, "Analysis of target detection performance for wireless sensor networks," in *Proc. DCOSS '05*, pp. 276–292, Marina del Rey, CA, Jul 2005.

[6] R. Catanuto, S. Toumpis, and G. Morabito, "Opti {c,m} al: Optical/optimal routing in massively dense wireless networks," in *Proceedings of IEEE INFOCOM '07*, pp. 1010–1018, Anchorage, Alaska, May 2007.

[7] C. Chiasserini, R. Gaeta, M. Garetto, M. Gribaudo, D. Manini, and M. Sereno, "Fluid models for large-scale wireless sensor networks," *Performance Evaluation*, vol. 64, no. 7-8, pp. 715–736, Aug 2007.

[8] R. Cruz, "A calculus for network delay. i. network elements in isolation," *IEEE Trans. on Information Theory,*, vol. 37, no. 1, pp. 114–131, Jan 1991.

[9] O. Dousse, C. Tavoularis, and P. Thiran, "Delay of intrusion detection in wireless sensor networks," in *Proc. ACM MobiHoc '06*, pp. 155–165, Florence, Italy, May 2006.

[10] O. Dousse, P. Thiran, and M. Hasler, "Connectivity in ad-hoc and hybrid networks," in *Proc. IEEE INFOCOM*, pp. 1079–1088, New York, Jun 2002.

[11] E. Duarte-Melo and M. Liu, "Data-gathering wireless sensor networks: Organization and capacity," *Computer Networks*, vol. 43, pp. 519–537, Nov 2003.

[12] E. Duarte-Melo, M. Liu, and A. Misra, "A modeling framework for computing lifetime and information capacity in wireless sensor networks," in *WiOpt '04*, Cambridge, UK, Mar 2004.

[13] M. Fidler, "An end-to-end probabilistic network calculus with moment generating functions," in *Proc. IEEE IWQoS*, pp. 261–270, New Haven, CT, Jun 2006.

[14] M. Franceschetti, O. Dousse, D. Tse, and P. Thiran, "Closing the gap in the capacity of wireless networks via percolation theory," *IEEE Transactions on Information Theory*, vol. 53, no. 3, pp. 1009–1018, Mar 2007.

[15] R. G. Gallager, *Discrete Stochastic Processes*. Kluwer Academic Publishers, 1995.

[16] M. Gribaudo, D. Manini, A. Nordio, and C. Chiasserini, "Analysis of IEEE 802.15.4 sensor networks for event detection," in *Proc. IEEE Globecom '09*, pp. 152–157, Honolulu, Hawaii, Nov 2009.

[17] L. Gu, D. Jia, P. Vicaire, T. Yan, L. Luo, A. Tirumala, Q. Cao, T. He, J. Stankovic, T. Abdelzaher, and B. H. Krogh, "Lightweight detection and classification for wireless sensor networks in realistic environments," in *Proc. ACM SenSys '05*, San Diego, CA, Nov 2005.

[18] V. C. Gungor, O. B. Akan, and I. F. Akyildiz, "A real-time and reliable transport $(RT)^2$ protocol for wireless sensor and actor networks," *IEEE/ACM Trans. on Networking*, vol. 16, no. 2, pp. 359–370, Apr 2008.

[19] P. Gupta and P. R. Kumar, "The capacity of wireless networks," *IEEE Trans. on Information Theory*, vol. IT-46, no. 2, pp. 388–404, Mar 2000.

[20] J. Kim, X. Lin, and N. Shroff, "Optimal anycast technique for delay-sensitive energy-constrained asynchronous wireless sensor networks," in *Proc. IEEE INFOCOM '09*, pp. 412–421, Rio de Janeiro, Brazil, Apr 2009.

[21] S. Kulkarni, A. Iyer, and C. Rosenberg, "An address-light, integrated mac and routing protocol for wireless sensor networks," *IEEE/ACM Trans. on Networking*, vol. 14, no. 4, pp. 793–806, Aug 2006.

[22] V. G. Kulkarni, "Fluid models for single buffer systems," *Frontiers in Queueing: Models and Applications in Science and Engineering*, pp. 321–338, 1997.

[23] P. Levis, N. Lee, M. Welsh, and D. Culler, "TOSSIM: accurate and scalable simulation of entire tinyos applications," in *Proc. ACM SenSys '03*, pp. 126–137, Los Angeles, CA, Nov 2003.

[24] S. Liu, K.-W. Fan, and P. Sinha, "CMAC: An energy efficient mac layer protocol using convergent packet forwarding for wireless sensor networks," in *Sensor, Mesh and Ad Hoc Communications and Networks, 2007. SECON '07. 4th Annual IEEE Communications Society Conference on*, pp. 11–20, Jun 2007.

[25] Y. Liu, F. Lo Presti, V. Misra, D. Towsley, and Y. Gu, "Fluid models and solutions for large-scale IP networks," in *Proc. ACM SIGMETRICS '03*, pp. 91–101, San Diego, CA, Jun 2003.

[26] D. Marco, E. J. Duarte-Melo, M. Liu, and D. L. Neuhoff, "On the many-to-one transport capacity of a dense wireless sensor network and the compressibility of its data," in *Proc. IPSN'03*, Palo Alto, CA, Apr 2003.

[27] W. Pak, J.-G. Choi, and S. Bahk, "Tier based anycast to achieve maximum lifetime by duty cycle control in wireless sensor networks," in *Wireless Communications and Mobile Computing Conference, 2008. IWCMC '08. International*, pp. 123–128, Aug 2008.

[28] J. Polastre, J. Hill, and D. Culler, "Versatile low power media access for wireless sensor networks," in *Proc. ACM SenSys '04*, Baltimore, MA, Nov 2004.

[29] S. Toumpis and L. Tassiulas, "Packetostatics: Deployment of massively dense sensor networks as an electrostatics problem," in *Proc. IEEE INFOCOM '05*, vol. 4, no. 2005, Miami, FL, Mar 2005.

[30] M. C. Vuran and I. F. Akyildiz, "Xlp: A cross-layer protocol for efficient communication in wireless sensor networks," *IEEE Transactions on Mobile Computing*, vol. 9, no. 11, Nov 2010.

[31] Y. Wang, M. C. Vuran, and S. Goddard, "Cross-layer analysis of the end-to-end delay distribution in wireless sensor networks," in *Proc. IEEE RTSS '09*, Washington, DC, Dec 2009.

[32] ——, "Stochastic analysis of energy consumption in wireless sensor networks," in *Proceedings of the 7th IEEE Communications Society Conference on Sensor, Mesh and Ad Hoc Communications and Networks (SECON 2010)*, Boston, MA, Jun 2010.

[33] Y. Xue, M. C. Vuran, and B. Ramamurthy, "Cost-efficiency of anycast-based forwarding in duty-cycled wsns with lossy channel," in *Proc. IEEE SECON 2010*, Boston, MA, Jun 2010.

[34] J. Zhang, G. Zhou, S. Son, J. Stankovic, and K. Whitehouse, "Performance analysis of group based detection for sparse sensor networks," in *Proc. ICDCS 2008*, pp. 111–122, Jun 2008.

[35] M. Zúniga and B. Krishnamachari, "An analysis of unreliability and asymmetry in low-power wireless links," *ACM Trans. on Sensor Networks*, vol. 3, no. 2, Jun 2007.