Networked Wireless Sensor Data Collection: Issues, Challenges, and Approaches

Feng Wang, *Student Member, IEEE,* and Jiangchuan Liu, *Senior Member, IEEE*

***Abstract*—Wireless sensor networks (WSNs) have been applied to many applications since emerging. Among them, one of the most important applications is *Sensor Data Collections*, where sensed data are collected at all or some of the sensor nodes and forwarded to a central base station for further processing. In this paper, we present a survey on recent advances in this research area. We first highlight the special features of sensor data collection in WSNs, by comparing with both wired sensor data collection network and other WSN applications. With these features in mind, we then discuss the issues and prior solutions on the utilizations of WSNs for sensor data collection. Based on different focuses of previous research works, we describe the basic taxonomy and propose to break down the networked wireless sensor data collection into three major stages, namely, the *deployment* stage, the *control message dissemination* stage and the *data delivery* stage. In each stage, we then discuss the issues and challenges, followed by a review and comparison of the previously proposed approaches and solutions, striving to identify the research and development trend behind them. In addition, we further discuss the correlations among the three stages and outline possible directions for the future research of the networked wireless sensor data collection.**

***Index Terms*—Wireless sensor network, sensor data collection, deployment, data gathering, message dissemination.**

I. INTRODUCTION

WIRELESS sensor networks have been applied to many applications since emerging [1]. Among them, one of the most important applications is sensor data collection, where sensed data are continuously collected at all or some of the sensor nodes and forwarded through wireless communications to a central base station for further processing. In a WSN, each sensor node is powered by a battery and uses wireless communications. This results in the small size of a sensor node and makes it easy to be attached at any location with little disturbances to the surrounding environment. Such flexibility greatly eases the costs and efforts for deployment and maintenance and makes wireless sensor network a competitive approach for sensor data collection comparing with its wired counterpart. In fact, a wide range of real-world deployments have be witnessed in the past few years. Examples are across wildlife habitat monitoring [2], environmental research [3][4], volcano monitoring [5][6], water monitoring [7], civil engineering [8][9] and wildland fire forecast/detection [10], to name but a few.

The unique features of WSNs, however, also bring many new challenges. For instance, the lifetime of a sensor node is constrained by the battery attached on it, and the network lifetime in turn depends on the lifetime of sensor nodes, thus to further reduce the costs of maintenance and redeployment, the consideration of energy efficiency is often preferred in a WSN design [11]. Moreover, these challenges are complicated by the wireless losses and collisions during sensor nodes communicate with each other.

On the other hand, the requirements specified by sensor data collection applications also raise issues that need to be considered in the network design. First of all, the deployed sensors may need to cover the full area that the sensor data collection application is interested in. And to acquire data accurately, sensors may be required to be put at specific locations. Also different types of data (temperature, light, vibration) may be obtained by different sensors with different sampling rates. These issues may cause unbalanced energy consumptions over a WSN and significantly shorten the network lifetime if not handling carefully. In addition, since data are required to be delivered to the base station without any information loss, the data aggregation/fusion operations [12] are hard to be applied, which calls for novel solutions for enhancing the network performance.

In this paper, we present a survey on recent advances of tackling these challenges. By comparing with both wired sensor data collection networks and other applications of WSNs, we first highlight the special features of sensor data collections in WSNs. With these features in mind, we then discuss issues and prior solutions on sensor network deployment and data delivery protocols. In addition, we discuss different approaches for control message dissemination, which acts as an indispensable component for network control and management and can greatly affect the overall performance of WSNs for sensor data collections.

The remainder of this paper is organized as follows: In Section II, we compare WSNs for sensor data collection with the wired sensor data collection networks and WSNs for other applications, aiming to highlight the special features to be considered in the network design. Section III presents a detailed investigation on different deployment strategies and Section IV discusses issues and solutions on the data delivery protocol design. Prior mechanisms on message dissemination for network management and control are investigated in Section V. Finally, Section VI concludes the paper and gives further discussions on the directions of future work.

Feng Wang and Jiangchuan Liu are with the School of Computing Science, Simon Fraser University, British Columbia, Canada (email: {fwa1, jcliu}@cs.sfu.ca).

II. OVERVIEW

A. Wireless Sensor Networks

As a type of newly emerged network, WSN has many special features comparing with traditional networks such as Internet, wireless mesh network and wireless mobile ad hoc network. First of all, a sensor node after deployed is expected to work for days, weeks or even years without further interventions. Since it is powered by the attached battery, high efficient energy utilization is necessary, which is different from Internet as well as wireless mesh and mobile ad hoc network, where either constant power sources are available or the expected lifetime is several order of magnitude lower than it is for WSNs.

Although a sensor node is expected to work through a long time, it is often not required to work all the time, i.e., it senses ambient environment, processes and transmits the collected data; it then idles for a while until the next sensing-processing-transmitting cycle. To support fault tolerance, a location is often covered by several sensor nodes. To avoid duplicate sensing, while one node is performing the sensing-processing-transmitting cycle, other nodes are kept in the idle state. In these cases, the energy consumption can be further reduced by letting the idle nodes turn to *dormant* state, where most of the components (e.g., the wireless radio, sensing component and processing unit) in a sensor node are turned off (instead of keeping in operation as in the idle state). When the next cycle comes (indicated by some mechanism such as an internal timer), these components are then waken up back to the normal (*active*) state again. Define *duty-cycle* as the ratio between active period and the full active/dormant period. A low duty-cycle WSN clearly enjoys a much longer lifetime for operation. This feature has been exploited in quite a few research works [13][14]. However, as will be shown later in this paper, the new working pattern also brings challenges to the network design.

Another special feature related to energy consumption is to control the transmission range of a sensor node. Previous researches have shown that one of the major energy costs in a sensor node comes from the wireless communication, where the main cost increases with the 2 to 6 power of the transmission distance [15][16]. As a result, the transmission range of a sensor node is often preferred to be adjustable and may be dynamically adjusted to achieve better performance and lower energy consumption.

B. Sensor Data Collection

In a sensor data collection application, sensors are often deployed at the locations specified by the application requirement to collect sensing data. The collected sensing data are then forwarded back to a central base station for further processing. Traditionally, these sensors are connected by wires which are used for data transmission and power supply. However, the wired approach is found to need great efforts for deployment and maintenance. To avoid disturbing the ambient environment, the deployment of the wires has to be carefully designed. And a breakdown in any wire may cause the whole network out of service and enormous time and efforts may be taken to find out and replace the broken line. In addition, the sensing environment itself may make the wired deployment and its maintenance very difficult, if not impossible. For example, the environment near a volcano [5][6] or a wildfire scene [10], where the hot gases and steams can damage a wire easily. Indeed, even in a less harsh environment like wild habitat [2][3][4] or a building [17][9][7], the threats from rodents are still critical and make the protection of wires much more difficult than that of sensors. All these issues make wireless sensor network a pleasant choice as it emerges with technology advances.

On the other hand, although many research efforts have been done on WSNs, and quite a few prototype or preliminary systems have been deployed, sensor data collection in WSNs is still in its early stage and its special features call for novel approaches and solutions different from other applications.

For example, a common work pattern in most of other applications, such as target tracking [18], is that sensing data or information are locally processed and stored at some nodes and may be queried later by some other nodes [19]. Sensor data collection, nevertheless, requires all sensing data are correctly and accurately collected and forwarded to the base station, since the processing of these data needs the global knowledge and is much more complex than that in other applications like target tracking. This feature also prevents using data aggregation/fusion techniques to enhance the network performance. As a result, the major traffic in sensor data collection is the reported data from each sensor to the base station. Such “many-to-one” traffic pattern, if not carefully handled, will cause high unbalanced and inefficient energy consumption in the whole network. As a concrete example, the energy hole problem was reported and discussed in [16], where sensor nodes close to the base station are depleted quickly due to traffic relays and create a hole shape area that leaves the remaining network disconnected from the base station. One possible solution to alleviate such issues is using mobile entities that proactively move around and collect data in the sensing field [20][21]. However, due to the harshness of the sensing environment as well as to minimize the disturbances, such a solution is often unfeasible in the context of sensor data collection.

In addition, unlike other WSNs, the sensors used in sensor data collection are often in great amount and of different types [2][4][8][9], from traditional thermometer, hygrometer to very specialized accelerometer and strain sensor. These sensors work at their own sample rates specified by the applications, and the rates may be different from one to another, e.g. a typical sampling rate of an accelerometer is $100Hz$, while the frequency to sample temperature is much lower. Such difference in turn leads to different transmission rates to relay data from different type of sensors, which may further aggravate the unbalance of the traffic pattern and energy consumption and thus result in performance inefficiencies.

C. Taxonomy

In practice, using WSNs for sensor data collection can be broken into three major stages, namely, the *deployment*

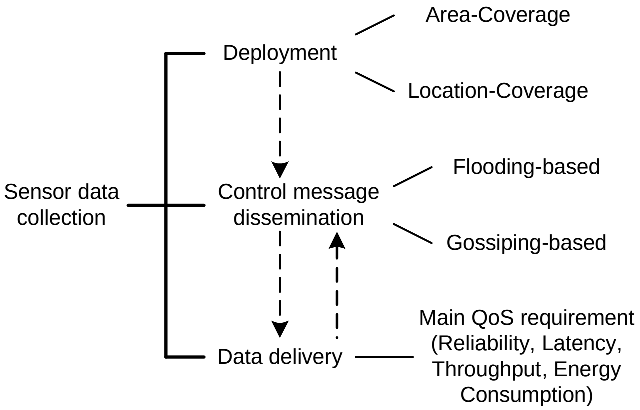


Fig. 1. Major stages and taxonomy of using wireless sensor networks for sensor data collection.

stage, the *control message dissemination* stage and the *data delivery* stage, and each stage has its own issues and focuses. Fig. 1 illustrates the three stages. The deployment stage addresses the issues such as how to deploy the network in the sensing field. Based on the application requirement, the problem can be further categorized into the area-coverage deployment and the location-coverage deployment, where the former requires each location within the sensing field must be covered by some sensor nodes and the latter requires the sensor nodes must be attached to some locations specified by the applications. In the control message dissemination stage, network setup/management and/or collection command messages are disseminated from the base station to all sensor nodes, where the challenges lie in how to disseminate messages to all the sensor nodes with small transmission costs and low latencies. Flooding and gossiping are two commonly used dissemination approaches that can be easily adopted in WSNs. Thus although their basic forms are known inefficient, later works have enhanced them with improved efficiency while retaining their robustness in the presence of error-prone wireless transmissions. The data delivery stage fulfills the main task of sensor data collection. Based on the information indicated by stage 2, sensed data are gathered at different sensor nodes and delivered to the base station, where different QoS requirements from the applications will infer different approach designs with different main QoS considerations. It is worth noting that stage 2 and stage 3 may serve alternatively, so that after one round data collection, new setup/command messages are disseminated and thus start a new round of collection. In the following sections, we will investigate these stages one by one in detail on their recent progresses and discuss potential directions for the future work.

III. DEPLOYMENT STRATEGIES

The first step for designing a WSN is to consider how it is deployed in the sensing environment. Based on different application requirement, different deployment strategies may be applied. For sensor data collection, one typical requirement is *area-coverage*, where each location within the sensing field must be covered by at least k ($k \geq 1$) sensor nodes [22], and the main purpose of $k > 1$ is for fault tolerance. Yet another kind of typical requirement is *location-coverage*, where sensor nodes must be attached to some specific locations that are chosen carefully by applications [23][24]. In the following, we will investigate different deployment problems and the resulting solutions, which were proposed to achieve different requirements.

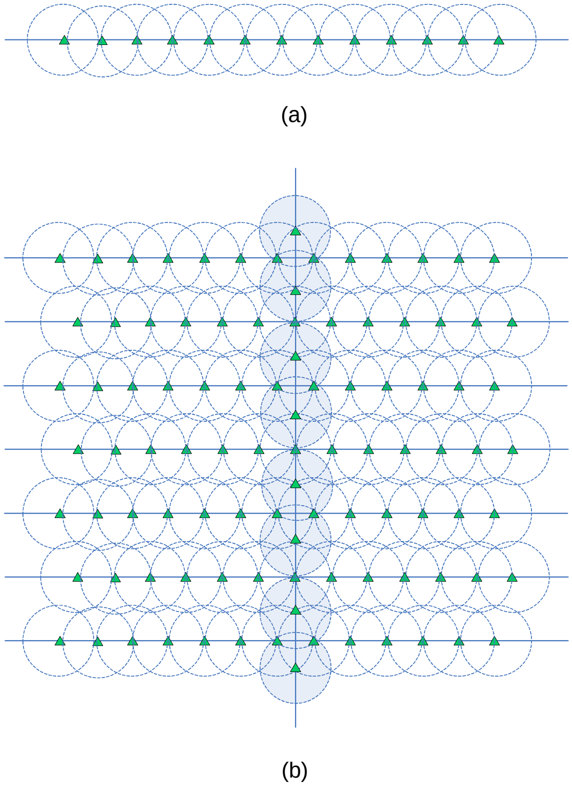


Fig. 2. The strip-based pattern for full coverage and 1-connectivity [25]. (a) shows a strip; and (b) is the deployment strategy using strips as building blocks with a vertical line of sensors to meet the connectivity requirement. Sensor nodes are denoted by small triangles and each dashed circle shows the sensing range of the sensor node at its center.

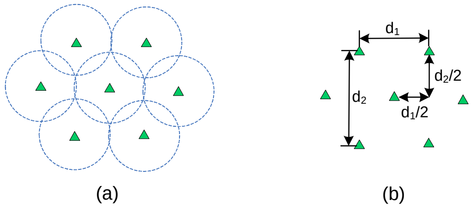


Fig. 3. The diamond pattern for full coverage and 4-connectivity [26].

A. Deployment for Area-Coverage

For area coverage requirement, where each location within the sensing field must be covered by at least k ($k \geq 1$) sensor nodes, one solution is using random deployment, which is widely adopted in other WSN applications such as target tracking [18]. An advantage of random deployment is that sensor nodes can be deployed by spraying from airplanes or simply scattering with moderate human efforts. Yet, an issue here is that how many sensor nodes are required so as to achieve the k-coverage requirement.

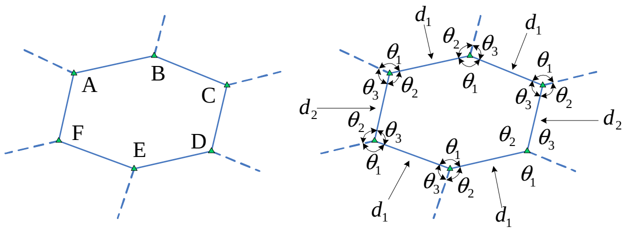


Fig. 4. The universally element pattern for full coverage and k-connectivity ($k \leq 6$) [27].

Given that sensor nodes are distributed as a Poisson point process, the authors of [28] derived the required density to fulfill k-coverage. In particular, the authors assumed that each sensor node is independently deployed at uniform distribution in the $l \times l$ sensing field R, and for any sub-region A, the node number in A follows a Poisson distribution with the average as $\lambda\|A\|$, where the density $\lambda = n/\|R\|$ with n denoting the total number of sensor nodes. Let $\chi_k(x)$ denote the indication function of whether a point x is covered by at most $k-1$ sensor nodes, i.e.,

$$\chi_k(x) = \begin{cases} 1, & \text{if at most } k-1 \text{ nodes cover point } x; \\ 0, & \text{otherwise.} \end{cases} \tag{1}$$

Then given the deployment following Poisson point process and each sensor node covers a unit disk area centered at the node itself, the probability of $\chi_k(x) = 1$ can be computed by the sum of the probabilities that the point x is covered by $0, 1, \ldots, k-1$ sensor nodes, which follows

$$P(\chi_k(x) = 1) = \sum_{i=0}^{k-1} \left(\frac{e^{-\lambda}\lambda^i}{i!} \right) = e^{-\lambda} \left(\sum_{i=0}^{k-1} \frac{\lambda^i}{i!} \right) . \tag{2}$$

Let V_k denote the region that is covered by at most $k-1$ nodes, which can be calculated by counting the calculus of each point x that is covered by at most $k-1$ sensor nodes (i.e., $\chi_k(x) = 1$) as shown below

$$V_k = \int_R \chi_k(x) dx . \tag{3}$$

Then to make the whole area be covered by at least k nodes, we need to find a suitable density λ that lets $P(V_k = 0) \rightarrow 1$ as $l \rightarrow +\infty$. And the authors of [28] proved that the following equation is a tight bound to fulfills this requirement:

$$\lambda = \log l^2 + (k+1) \log \log l^2 + c(l) \tag{4}$$

with $c(l) \rightarrow +\infty$ as $l \rightarrow +\infty$.

Later, with the consideration of boundary effects, the authors of [29] proposed that by deploying sensor nodes on regular grids, grid deployment renders asymptotically lower node density than random deployment. To consider boundary effects, the authors divide the sensing field into small grids with equal side length, which are further categorized into inner grids, side grids and corner grids based on their distances to the borders of the sensing field. The probability for each type of grids not being k-covered is then calculated by a technique similar to [28] as described in the previous paragraph. The results show that for grid deployment,

$$\begin{aligned} &(-\log(1-p))\lambda = \\ &\log l^2 + 2k \log \log l^2 + 2\sqrt{-2\pi \log l^2 \log(1-p)} + c(l) \end{aligned} \tag{5}$$

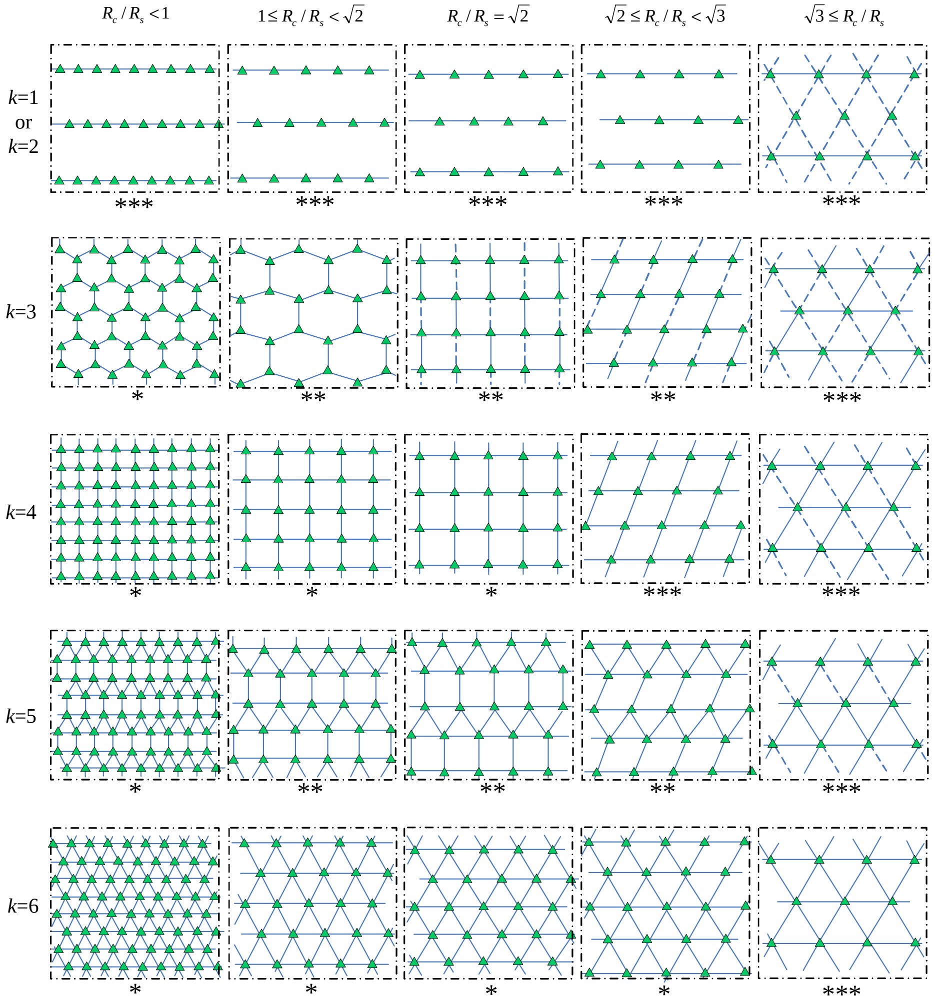
is a tight upper bound; while for Poisson point (or uniform) random deployment, the bound is

$$p\lambda = \log l^2 + 2k \log \log l^2 + c(l) , \tag{6}$$

where p is the probability that a sensor node is in active state. Intuitively, if p is a constant with $0 < p \leq 1 - \epsilon$ for any constant ϵ ($0 < \epsilon < 1$), we have $p < -\log(1-p)$; and as $l \rightarrow +\infty$, $\log l^2$ dominates the right sides of both Eqns. (5) and (6), which indicates the density required by Poisson point (or uniform) random deployment will be higher. This conclusion proposed in [29] implies an advantage of manual deployment on reducing equipment costs, since currently the price of a sensor node is still far from negligible.

Besides the coverage requirement, another important issue for WSN deployment is connectivity, since if the network is partitioned, an entire portion of the network becomes useless, i.e., the sensing data can not reach the base station. If the communication range R_c is at least twice of its sensing range R_s ($R_c \geq 2R_s$), then the full coverage of a region also implies the full connectivity [30]. Otherwise, the connectivity need to be explicitly considered. Assuming $R_c = R_s$, [25] proposed a strip-based deployment pattern to reduce the required number of sensor nodes. Fig. 2 gives an illustration, where on a strip, the gap between two neighboring sensors is set to the maximum value that can still meet the connectivity requirement, and the strips are then deployed at the maximum in-between distance that still fulfills the coverage requirement, with a vertical line of sensors to keep the connectivity requirement. In [31], this strip-based deployment was shown to be asymptotically optimal for both full coverage and 1-connectivity when $R_c/R_s < \sqrt{3}$. In addition, the authors of [31] extended the strip-based deployment a step further by adding another vertical line of sensors and showed the extension is optimal for both full coverage and 2-connectivity.

Later, a diamond pattern was explored in [26] and shown to be asymptotically optimal to achieve full coverage and 4-connectivity. Fig. 3 shows an example of the diamond pattern, where d_1 and d_2 are two parameters determined by R_c and R_s. The optimal deployment patterns for full coverage and k-connectivity with $k \leq 6$ was finally completed in [27], where the authors proposed a universally element pattern that unifies prior results for 1-, 2- and 4-connectivity, as well as their results for 3-, 5- and 6-connectivity. The proposed element pattern is "universally" in terms that all previously known optimal patterns of different connectivity (≤ 6) with full coverage can be generated by repeating certain specific forms of the pattern. Fig. 4 and Fig. 5 illustrate the universally element pattern proposed in [27] and the resulting complete set of optimal patterns for $k \leq 6$ respectively, where a γ-optimal deployment means the deployment is optimal among



$$\begin{cases} \theta_1 = \pi, \\ \theta_2 = \min(\frac{2\pi}{3}, \arccos(\frac{R_c}{2R_s + 2\sqrt{R_s^2 - \frac{R_c^2}{4}}})), \\ d_1 = \min(R_c, \sqrt{3}R_s), \\ d_2 = \left((\frac{d_1}{2})^2 + (\sqrt{R_s^2 - (\frac{d_1}{2})^2} + R_s)^2 \right)^{\frac{1}{2}}. \end{cases}$$

$$\begin{cases} \theta_1 = \min(\pi, \max(\frac{2\pi}{3}, 4\arcsin\frac{R_c}{2R_s})), \\ \theta_2 = \max(\min(\frac{2\pi}{3}, 2\arccos\frac{R_c}{2R_s}), \min(\frac{2\pi}{3}, 2\arcsin\frac{R_c}{2R_s})), \\ d_1 = d_2 = \min(R_c, \sqrt{3}R_s). \end{cases}$$

$$\begin{cases} \theta_1 = \pi, \\ \theta_2 = \min(\frac{2\pi}{3}, \max(2\arcsin\frac{R_c}{2R_s}, \frac{\pi}{2})), \\ d_1 = d_2 = \min(R_c, \sqrt{3}R_s). \end{cases}$$

$$\begin{cases} \theta_1 = \frac{\pi}{3}, \\ \theta_2 = \frac{\pi}{3} + \min(\frac{2\pi}{3}, \max(2\arcsin\frac{R_c}{2R_s}, \frac{\pi}{2})), \\ d_1 = d_2 = \min(R_c, \sqrt{3}R_s). \end{cases}$$

$$\begin{cases} \theta_1 = \pi, \\ \theta_2 = \frac{2\pi}{3}, \\ d_1 = d_2 = \min(R_c, \sqrt{3}R_s). \end{cases}$$

* Conjectured Globally Optimal (proving or disproving is unknown) ** γ-Optimal (Conjectured Globally Optimal) *** Globally Optimal

Fig. 5. A complete set of optimal patterns achieving full coverage and k-connectivity with $k = 1, \ldots, 6$, respectively (where the sensing range R_s is invariant and the communication range R_c varies) [27]. These patterns are specific forms of the universally element pattern (shown in Fig. 4) defined by expressions of θ_1, θ_2, ($\theta_3 = 2\pi - \theta_1 - \theta_2$), d_1 and d_2 on the right side of the above deployment patterns. Note that there are one and two vertical lines of nodes for global connectivity in 1- and 2-connectivity patterns, respectively. They are not shown for the sake of simplicity.

all regular deployments[1].

Even with connectivity considered at the initial stage, as time goes on, some sensor nodes may consume more energy than others due to more traffic relaying. This leads to unbalanced energy costs and the network being partitioned prematurely with a great number of nodes still having a large amount of energy. To alleviate this problem, the authors of [32] have proposed to deploy additional relay nodes so as to take the burden of traffic relaying from sensor nodes and prolong the lifetime of the whole network. In addition, they proposed a hybrid approach to deploy relay nodes while considering the connectivity and network lifetime simultaneously. Specifically, the sensing field is divided into three parts based on the distance from the base station. The inner part is the part closest to the base station, where relay nodes can reach the base station by one hop communication. The outer part is the part farthest to the base station, where no traffic from other relay nodes needs to be relayed and the relay nodes only relay traffic directly from the sensor nodes. The medium part is the part remaining between the inner and outer parts, where relay nodes need to relay traffics from both the sensor nodes and the relay nodes one hop farther from the base station. Different relay node density is then derived for each of the three parts. Based on the results, the authors of [32] suggested to divide relay nodes into two portions. The first portion is distributed proportionally to the derived density for each part, and the remaining relay nodes are then deployed to compensate for connectivity, i.e., to guarantee a relay node is within the communication range of a sensor node with high probability.

The proposed strategy, however, may become too coarse as the sensing field grows large, as the size of the medium part will increase with the sensing field. Intuitively, one possible solution is to further divide the medium part into sub-parts

---

[1] In a regular deployment, all vertices have the same degree, and for each vertex, the angles formed by neighboring edges can be numbered in some order such that if two angles from different vertices are numbered the same, they are also of the same degrees.

to enable a finer approach with different density for each sub-parts. On the other hand, an interesting direction would be to explore solutions by manually deploying relay nodes to compensate the unbalanced energy consumption, which is expected to provide useful strategies and bounds as guidelines for practical deployment.

B. Deployment for Location-Coverage

Another typical coverage requirement for sensor data collection is that sensors are manually attached to some specified locations that are carefully chosen by applications. One example is the project conducted on TsingMa Bridge in Hong Kong [23], where the bridge is equipped with a large number of accelerometers, thermometers and strain sensors to monitor its working conditions. Another recent project, which is still ongoing, is on the Guangzhou New TV Tower [24] in Guangzhou, China, where the tower will be attached with similar sensors for real-time monitoring and analyzing. In these systems, sensors are deployed at specified locations to fulfill the civil engineering requirements. Since the locations selected by applications are not necessarily considering the networking requirements such as connectivity and energy efficiency, additional relay nodes are often placed in the sensing field to match these requirements and facilitate sensing data deliveries from sensor nodes to the base station. Yet an issue is how many relay nodes are required and where to deploy them.

In [33], the authors modeled the relay node placement problem for connectivity as Steiner Minimum Tree with Minimum number of Steiner Points and bounded length (SMT-MSP) problem [34] and proposed a 3-approximation algorithm. Specifically, considering in a graph, sensor nodes are the given vertices and relay nodes are steiner points[2], then the problem to use minimum number of relay nodes to connect all sensor nodes becomes to use minimum number of steiner points to connect all the given vertices, where the constraint is that the edge length can only be less than or equal to the wireless communication range. And the main idea of the proposed 3-approximation algorithm is to conduct the minimum spanning tree algorithm on the given vertices and insert an intermediate stage when the remaining edges between the given vertices are longer than the communication range. In the inserted stage, a steiner point (with three edges) is added to connect three connected components into one if the steiner point can connect each component with one edge whose length is less than or equal to the communication range. In addition, if an edge longer than the communication range being selected by the algorithm, minimum number of steiner points are also added on the edge to break it into smaller ones with length less than or equal to the communication range.

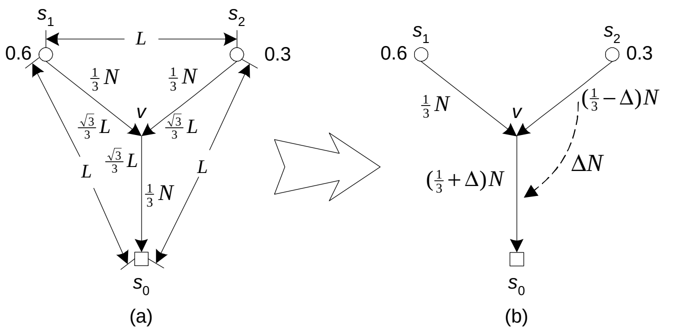


Fig. 6. An example of relay node deployment: (a) connectivity-based deployment; (b) traffic-aware deployment [38]. s_1, s_2 are sources with data rate of 0.6 and 0.3. s_0 is the base station. Given N relay nodes, by scheme (a) which only considers connectivity, nodes relaying the traffic from v to s_0 will die much earlier than those relaying from s_1 and s_2 to v, while by strategically deploying more nodes (ΔN) on section (v, s_0) (from less busy section (s_2, v)), the network lifetime is prolonged.

Later, to enable fault-tolerance, a series of approximation algorithms [35][36][37] have been proposed to place minimum number of relay nodes while achieving k-connectivity with $k \geq 2$. The core idea in these papers is to compute k-connected spanning subgraph from the full connected graph containing all sensor nodes as vertices. In addition, the edge between each pair of vertices is assigned a weight equal to the minimum number of relay nodes required to make any two neighboring nodes on the edge within each other's wireless communication range. Besides, some relay nodes are duplicated to avoid using sensor nodes to relay traffic. And redundant relay nodes are removed to reduce the costs.

Recently, it is noticed that for sensor data collection applications, only considering connectivity for relay node deployment may not always lead to the best performance in terms of the energy efficiency and network lifetime [38]. For example, in Fig. 6, by connectivity-based deployment (Fig. 6a), which is traffic oblivious, the optimal solution to maximize the network lifetime is to evenly distribute relay nodes along the minimum steiner tree topology. However, given the sensing data traffic from each sensor node to the base station, a better solution that considers such traffic patterns and moves some relay nodes from the low traffic edge to the high one (Fig. 6b) can further extend the network lifetime with more efficient energy utilization.

Motivated by this, the authors of [38] proposed a traffic-aware deployment strategy. In particular, given the number of relay nodes and the average sensing data rate at each sensor node, the authors modeled the traffic-aware deployment problem as a generalized Euclidian Steiner Minimum Tree problem (ESMT) [39], where sensor nodes are vertices and a number of steiner points are introduced so as to minimize the total length of the edges weighted by the rate of the aggregate traffic flowing through each edge. The authors proposed a hybrid algorithm to compute the number of required steiner points and their positions. And on each edge, a number of relay nodes can then be assigned proportionally to the amount of traffic passing through the edge.

---

[2] The concept of steiner points originates from the Steiner Minimum Tree problem. To minimize the total length of the edges that connect some given points, additional points may be introduced as intermediate points to connect other points and reduce the total length of the used edges. Here, the steiner points serve for similar purposes such as optimizing the total length of the used edges or reducing the length of each single edge to meet the edge length constraint.

Deployment Strategy	Coverage Requirement	Deploying Approach	Optimization Goal	Fault-Tolerance		Traffic-Aware
				Coverage	Connectivity	
Poisson Point Random [28]	Area	Random	Min. Node Number	k	-	No
Poisson Point/Uniform Random or Grid [29]	Area	Random/ Manual	Min. Node Number	k	-	No
Strip-based Pattern [25]	Area	Manual	Min. Node Number	1	1	No
Extended Strip-based Pattern [31]	Area	Manual	Min. Node Number	1	2	No
Diamond Pattern [26]	Area	Manual	Min. Node Number	1	4	No
Universally Element Pattern [27]	Area	Manual	Min. Node Number	1	k (≤ 6)	No
Lifetime/Connectivity Hybrid [32]	Area	Random	Max. Network Lifetime	1	1	Yes
Minimum Spanning Tree based Approximation [33]	Location	Manual	Min. Node Number	-	1	No
k-Connected Spanning Subgraph based Approximation [35][36][37]	Location	Manual	Min. Node Number	-	k	No
Traffic-Aware [38]	Location	Manual	Max. Network Lifetime	-	1	Yes

TABLE I
DIFFERENT DEPLOYMENT STRATEGIES, WHERE "-" MEANS NOT CONSIDERED OR NOT APPLICABLE.

C. Summary

Tab. I summarizes the deployment strategies that have been investigated in this section. We can see that for the deployment of area-coverage, both random and manual deploying approaches can be used, and the research trend starts from focusing on coverage only, then moves to combining coverage and connectivity together, and now is considering coverage and connectivity jointly with traffic-awareness. For the deployment of location-average, since sensor nodes must be placed at the specified locations precisely, the manual deploying approach becomes the only choice. Nevertheless, a similar pattern of the research trend can still be observed.

Following these prior works, there are still some open issues to be considered. First, although previous works assumed a flat 2-D sensing field for sensor deployment, practical sensing environments are often uneven and may be a 3-D structure (e.g. a building). In addition, there may be obstacles within the sensing field, which may reduce the range of transmissions crossing by or even prevent nodes from communication at all. Besides, as discussed in [40], sensor nodes may exhibit different sensing capabilities and within the sensing field, sensing requirements may vary at different locations. And the deployment may be required to fulfill multiple objectives, such as achieving high quality of monitoring, connectivity and long lifetime simultaneously [41]. How to optimally deploy sensor/relay nodes to address these issues are still under exploring. Another interesting issue is fault-tolerance, which has been considered individually either for coverage requirement or for connectivity requirement. However, in practice, both sensor and relay nodes are prone to failure due to the battery limitation and hash environment, and failing to fulfill either coverage requirement or connectivity requirement may lead to a premature termination of the network lifetime. Thus an important direction is to consider fault-tolerance jointly across all such requirements. In addition, how to integrate fault-tolerance with traffic-aware deployment is also an open question.

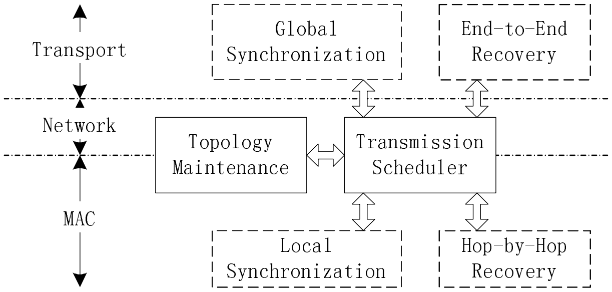


Fig. 7. A generic architecture of a data gathering approach. Mandatory components are shown by solid squares and optional components are shown by dashed squares.

IV. DATA DELIVERY APPROACHES

Given the deployment strategy of a WSN, the next step is to consider the data delivery approach, i.e, how to forward sensing data from each sensor node to the base station. Due to the "many-to-one" feature of the sensor data collection applications, the network topology is often considered as a tree topology rooted at the base station, which needs to be pre-defined or dynamically formed so that data packets can be routed along. On the other hand, the existence of wireless interferences and collisions makes the scheduling of data packet transmissions a challenging problem that needs to be carefully addressed to achieve effective and efficient accesses to the wireless medium. To this end, a cross-layer design is often involved, where the MAC, network and transport layer are considered together to achieve multiple goals such

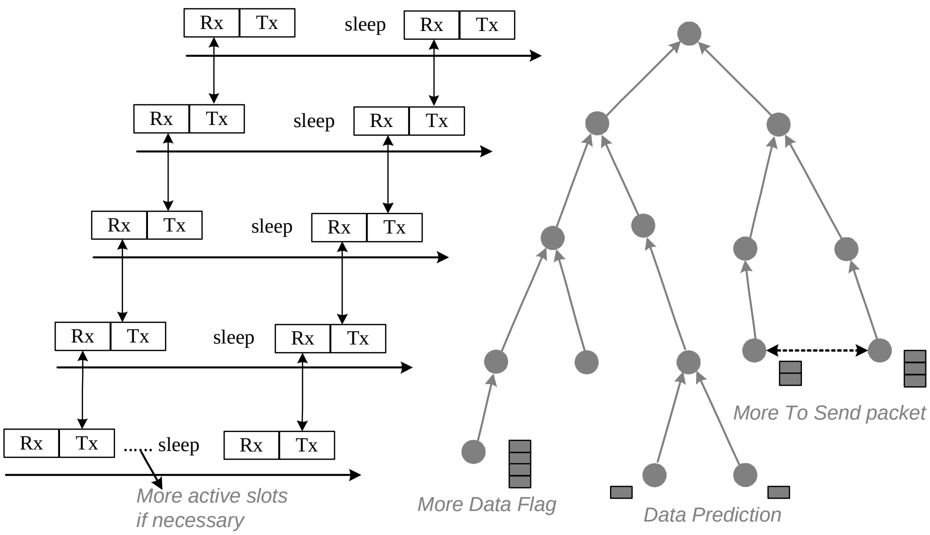
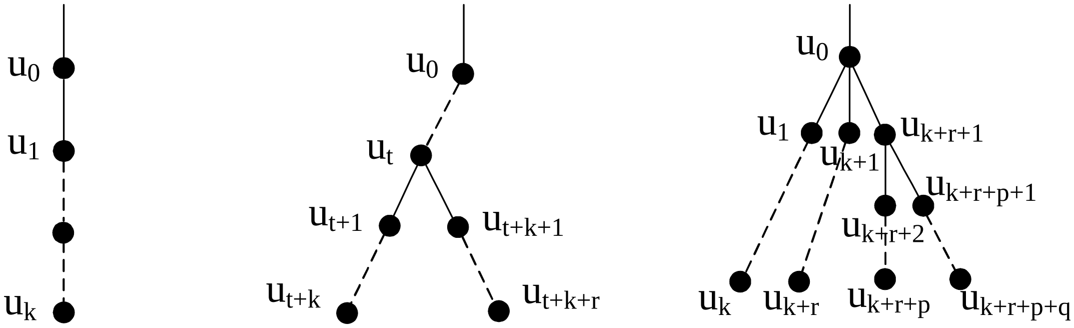


Fig. 8. An illustration of DMAC [42].



(a) linear graph (b) simple binary tree (c) another tree

Fig. 9. An illustration of time-optimal packet scheduling by STREE [43].

as energy efficiency as well as reliability. Fig. 7 illustrates a generic architecture for data delivery approaches. To collect data from sensor nodes, two mandatory components are topology maintenance and transmission scheduler. The topology maintenance component constructs a connected topology and maintains the connectivity during network dynamics and link quality variations. The transmission scheduler then schedules packet transmissions based on the information from other components so as to reduce collisions and energy wastes. Given different QoS requirements such as throughput, latency and reliability, different optional components may be added. Yet a more challenging issue is that sensor nodes are operating autonomously, thus the transmission scheduling algorithm needs to be designed to work in a distributed manner. In the following subsections, we will discuss recently proposed approaches by the categorization based on their major QoS considerations.

A. Reliability

One of the prior works [17] designed a WSN system named Wisden that adopted a data delivery approach with a stress on the reliability and exploited a hybrid scheme for reliable data deliveries using both hop-by-hop and end-to-end recoveries. Specifically, each node keeps tracking sequence numbers of packets it receives from a source node. A gap in the sequence numbers of received packets indicates packet loss. The sequence number of the missing packet and its source node ID are then stored in a missing list and piggy-backed when a packet is forwarded. The node that previously relayed the missing packet will then schedule a retransmission when it overhears the piggy-backed information. And to afford the retransmission in the hop-by-hop recovery, each newly received packet is cached for some short period. However, if heavy packet loss happens or the network topology changes due to dynamics such as link quality variations, the hop-by-hop recovery may fail due to the temporary overflow of missing lists or losing connections to previous forwarders. Thus an end-to-end recovery scheme is necessary to such situations. In particular, if a node overhears a piggy-backed missing list and finds some missing packets in the list sharing the same sources with those packets in its own packet cache, it then adds these packets into its own missing list and goes on to piggy-back their information in its transmissions. By this means, missing packet information will trace back hop-by-hop until reaching the sources. The sources will then re-send the packets and finish the circle of end-to-end recoveries.

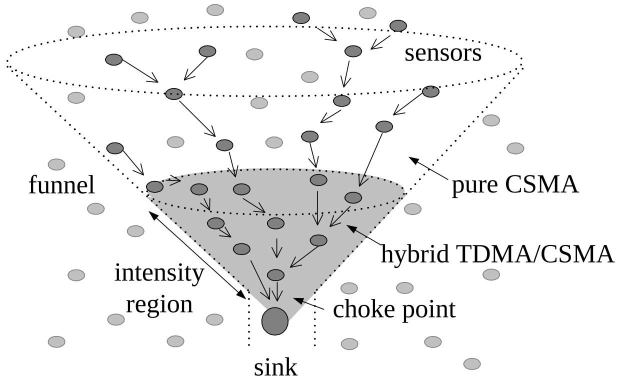


Fig. 10. An illustration of funneling-MAC [44].

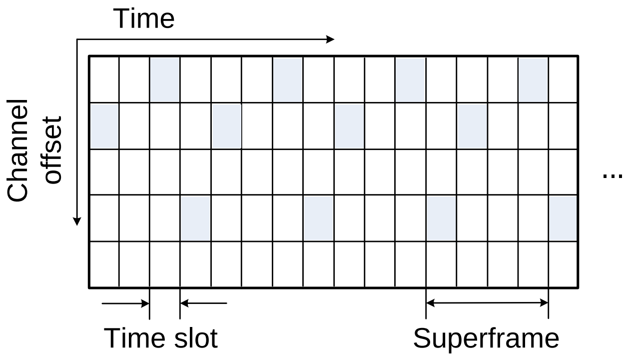


Fig. 11. An illustration of the matrix that divides the wireless space along both the time and channel dimensions [45]. The shaded slots are assigned to links by the centralized schedule. The schedule repeats as superframes and can be updated at the beginning of a superframe.

B. Latency

Since wireless communications consume a significant portion of energy budgets on sensor nodes, MAC protocols have been proposed to reduce idle listenings and turn the radio of the sensor node to sleep mode to save more energy. Such general designs, however, if being used for sensor data collection without careful consideration, may introduce extra latencies and even more energy costs. For example, if the next-hop neighbor is still sleeping, a node has to wait some extra time (called *sleeping latency*) until the neighbor turns active. On the other hand, to reduce sleeping latency, one approach is to let a node overhear for possible transmissions so as to temporarily increase its active duration for potential incoming packets. However, this would make all nodes that overhear a transmission spend extra time being active and consume more energy while only several of them really participate in the traffic relaying.

To reduce sleeping latency as well as energy costs, the authors of [42] proposed DMAC to enhance sensor data collection. The main idea is shown in Fig. 8. Based on the network topology, sensor nodes along a delivery path from a source node to the base station will turn to receiving, sending and sleep mode one after one in a sequential order. If there are more packets to send, a *More Data Flag* is piggy-backed with each previous packet to indicate the next transmission. The receiver then turns back to receiving mode, instead of sleep mode, to listen to the following packet. For the case that a receiver has more than one sender, on receiving a packet from one sender, the receiver predicts that there are packets from other senders and turns to receiving mode. And if nothing is heard, it turns back to sleep mode. In addition, within a transmission time slot, a contention-based mechanism (CSMA) is used for several senders to compete for one receiver, and another small time slot is reserved after each transmission slot for the failed sender to send a small *More To Send* packet, so as to make the receiver listen to its re-transmission instead of turning to sleep mode.

Another work named STREE was proposed in [43], which also targets on minimizing latency and reducing energy costs. By assuming global synchronization, time slot is defined to be the duration for successfully transmitting a maximum transmission unit. Within one time unit, a sensor node can sleep to save energy, or perform only one task of either sending or receiving. Given each sensor node has one packet to report to the base station during each round, for a linear topology as shown in Fig. 9(a), one optimal schedule to minimize the time duration for one round data collection is to let the even-level links and odd-level links be active alternatively, which is called *wavelike forwarding*. If there is any branch on the topology, as shown in Fig. 9(b), the optimal schedule can be achieved by letting the one path (e.g. $u_{t+k} \rightsquigarrow u_0$) does wavelike forwarding first, then after the branch ($u_{t+k} \rightsquigarrow u_{t+1}$) of the path is finished, the remaining part together with the other branch ($u_{t+k+r} \rightsquigarrow u_{t+k+1}$) will then form a new path and go on to do wavelike forwarding. In general, for any tree topology, an optimal schedule can be achieved by recursively applying wavelike forwarding to each branch. Let $N(u)$ denote the total number of nodes in the tree rooted at u. The authors showed that the time duration for all packets from the tree rooted at u to be forwarded up is $2N(u) - 1$. Furthermore, since the base station does not need to forward packets, it then can collect packets from two subtrees alternatively at the same time, e.g., in Fig. 9(c), if u_0 is the base station, link u_1u_0 and $u_{k+1}u_0$ can be active alternatively to send packets to u_0. Thus the optimal schedule can be achieved by letting all the subtrees of the base station do wavelike forwarding simultaneously and the base station collect packets from its children alternatively in descending order of subtree size. The time duration for one round data collection of the whole network is then derived as $\max(2N(u_1) - 1, N(u_0) - 1)$, where u_0 is the base station and u_1 is the child rooting the largest subtree.

Recently, it is noticed that a single piece of sensing data may be quite small and multiple pieces of data can still fit and be transmitted in one packet so as to reduce the transmission overhead [46]. Such batch transmission is different from traditional data aggregation/fusion techniques, where multiple data are combined into a smaller size at the price of losing original information. Since it is quite time-consuming to wait for enough data from one sensor node to form a packet and thus increases the latency, the authors of [46] proposed an approach named TIGRA to batch small sensing data from different sensor nodes into packets while these data are gathered along the collection tree. TIGRA uses a gathering mechanism similar to the wavelike forwarding introduced by STREE, but in a more general form. In particular, assuming each sensor node generates one sensing data and one packet can hold at most m sensing data, TIGRA let the links whose levels are at a distance of m be active together, such that data transferred by the links within distance m will eventually accumulate into the size of one packet and then be delivered to the base station. To make such a forwarding mechanism work for a general tree, TIGRA adopts a distributed graph coloring algorithm to resolve possible wireless collisions and interferences, where the links that interfere with each other are assigned by different colors so as to be active at different time slots to avoid collisions.

C. Throughput

As the main traffic in a WSN for sensor data collection is from all sensor nodes to the base station, the closer a sensor node is to the base station, the more packets it needs to relay. This will cause the funneling effect, as shown in Fig. 10, where the region close to the base station is heavily burdened and will experience significant collisions and packet losses if the MAC layer uses a CSMA-based protocol.

To solve this problem, Funneling-MAC [44] is proposed to improve the throughput of the network. The main idea is to adopt a TDMA protocol within the traffic intensity region (Fig. 10), which is assumed within the coverage of the base station's transmission power. By monitoring the arriving traffic from each path within the region, the base station assign time slots according to the traffic load. To keep synchronization, each time frame is started by a beacon from the base station, followed by the time slot assignment and then the time slots for packet transmissions. To facilitate emergency and control traffics, some time slots are reserved for transmissions by the CSMA protocol. In addition, the base station dynamically adjusts the size of the intensity region to exactly one hop smaller than the size that saturates all available time slots.

On the other hand, the authors of [47] focused on transport layer and proposed solutions to congestion control and fairness issues. Different from wired networks and other wireless networks, the congestion control in that paper is done by a per-hop manner. Given a routing tree topology, each node measures its average rate r at which packets can be sent. Then this r is evenly divided by the number of sources in the descendants of the node (including itself). The result is then compared with the rate assigned by its parent and the smaller one is selected and broadcast to its children if no congestion happens; otherwise, the selected rate is further decreased before being sent out. To achieve fairness, a node keeps the number of sources in the descendants of each child and uses these numbers as a weight to determine the packet from which child should be forwarded next. In addition, the authors of [47] proposed to use non-work conservation for queues and showed that although at the cost of throughput, non-work conservation helps to reduce the possibility of collisions and congestions.

D. Energy Consumption

In the general context of WSNs, a series of prior works have been proposed to reduce the energy consumption, where data aggregation/fusion techniques are mainly used to reduce the amount of traffics delivered towards the base station [15][49]. Due to the critical requirement for original data, such data aggregation/fusion techniques are unfeasible to be applied to sensor data collection, which calls for novel solutions to achieve good energy efficiencies.

Along this direction, the authors of [48] proposed Dozer, an ultra-low power data delivery scheme with a cross-design among MAC layer, topology control, routing and scheduling. To achieve this, the scheme adopts a TDMA protocol, where a beacon is broadcast at the beginning of each round, allocating time slots to possible transmissions within this round. During the tree topology construction and maintenance stage, nodes already integrated in the topology broadcast beacons and assign time slots for connection-requests from remaining nodes. Nodes receive the beacons then send connection-requests to one of the beacon senders and store the others locally for quick recoveries when current connections fail. During the data transmission stage, a parent node assigns each child a separate time slot for data reporting and local synchronization is achieved by letting all children listen to each beacon from the parent node. And by letting all nodes that are not listening or transmitting turn to sleep mode, a significant amount of energy can be saved. Besides, to counter collisions, a pseudo-random delay jitter is introduced before a beacon is broadcast at each round.

Recently, another approach named TSMP (Time Synchronized Mesh Protocol) is proposed to be used in sensor data collection [45]. Different from other previously mentioned works, TSMP advocates multi-channel communications[3] and a centralized network management. With its inherent global time synchronization mechanism supported, TSMP divides the wireless communication space along both the time and channel dimensions as illustrated in Fig. 11, where within one slot, a link transmits at most one packet with an ACK replied from the receiver indicating its reception. Each link or multiple non-interfering links then take one or more slots for packet transmissions based on the requirements of delivering packets from different sources to different destinations. To avoid wireless interferences and collisions, a centralized schedule that assigns slots to links is disseminated and then updated according to the dynamic variances. Sensor nodes only turn on their radios at slots where they are involved in packet transmissions according to the schedule. Since wireless interferences and collisions cause packet losses and retransmissions, a centralized schedule that resolves such interferences and losses would eventually reduce the energy consumption, which results in that TSMP can achieve more energy savings than the Dozer scheme does.

---

[3] In TSMP, the channel hopping technique is used for multi-channel communications, where the sender and receiver switch to different channel rapidly and occupy it for a short period (e.g. $10ms$ dwell period) before switching again.

Gathering Approach	Network Topology	Synchronization	Loss Recovery	Congestion Control	Sleep Mode	Rate Control	Source Rate	Main QoS Consideration
Wisden [17]	Tree	Not required	Hop-by-hop End-to-end	No	No	No	Any	Reliability
DMAC [42]	Tree	Local	Link layer	No	Yes	No	Any	Latency
STREE [43]	Tree	Global	Link layer	No	Yes	No	Single	Latency
TIGRA [46]	Tree	Global	Link layer	No	Yes	No	Single	Latency
Funneling-MAC [44]	Any	Area covered by Base Station	Link layer	Partial	No	Partial	Any	Throughput
Congestion Control and Fairness [47]	Tree	Not required	Link layer	Yes	No	Yes	Single	Throughput Fairness
Dozer [48]	Tree	Local	Hop-by-hop	Yes	Yes	Yes	Single	Energy Consumption
TSMP [45]	Any	Global	Hop-by-hop	Yes	Yes	Yes	Any	Energy Consumption

TABLE II
DIFFERENT DATA DELIVERY APPROACHES.

E. Summary

As sensor data constitute the major traffic in the network, wireless interferences and collisions are mostly occurring in the data delivery stage, which has more impacts on the MAC layer and thus generally leads to cross-layer designs among the MAC, network and transport layer as discussed in this section. Tab. II summarizes the data delivery approaches discussed in this section, where it is clear to see that based on different requirements, different components and mechanisms may be adopted for the final design. If the amount of the data being delivered to the base station is large, which means the transmission bandwidth is the bottle neck, the throughput then becomes the major concern and the transmission scheduler may also consider the rate/congestion control as well as the fairness. On the other hand, if the data amount is small and can not keep occupying the transmission bandwidth, turning off the radio to sleep mode is a good mechanism to reduce energy costs, but involving a tradeoff between the latency and energy consumption. In addition, based on different levels of the reliability requirement, choices can be made from the defaulted link layer recovery to the hop-by-hop recovery or even the end-to-end recovery, where more transmission overheads will be introduced to support a higher reliability.

Along these works, multiple tradeoffs among different QoS requirements can be considered jointly and explored further, which could be an interesting direction for future research. Also, most of prior works assume a single tree topology to be used for data delivery. On the other hand, as shown in Section III, most of the deployment topologies are more than a single tree structure and exploit multi-paths to provide fault-tolerance [31][26][27][35][36][37]. Thus how to integrate such inherent multi-paths provided by deployment with data delivery approaches to further enhance reliability is also available for exploring. Another issue is on energy saving and extending the network lifetime. Most of the previous works depended on turning sensor nodes into sleep mode to save energy and thus expect to extend the network lifetime. However, as mentioned in previous sections, the “many-to-one” traffic pattern in sensor data collection may cause high unbalance of energy consumption in the whole network and result in the premature termination of the network lifetime [32][38]. Thus how to balance energy consumption to extend the network lifetime while still keeping good energy efficiency is still an open question.

V. CONTROL MESSAGE DISSEMINATIONS

Till now we have discussed the deployment and data delivery issues of WSNs for sensor data collection. In such networks, there is another “one-to-many” traffic pattern where control messages such as network setup/management or collection commands are disseminated from the base station to all sensor nodes. Although such traffic is small in amount and generally causes less impacts on the MAC layer, it is still critical to the overall network performance. Previous research works largely overlooked such traffic or assumed it can be easily solved by existing broadcast approaches from wired or other types of wireless networks. Nevertheless, given the unique features of WSNs, necessity has been shown to call for novel solutions that can provide network-wide broadcast service with both energy-efficiency and reliability in this new context.

A. Basic Flooding and Gossiping

There have been numerous studies on broadcast in wired networks and in wireless ad hoc networks [50][51][52]. Among them, flooding and gossiping [53] are two commonly used broadcast approaches that can be easily adopted in WSNs. In flooding, each sensor node forwards the received message until the message reaches its maximum hop count. This approach provides high robustness against wireless communication loss and high reliability for message delivery. It however causes many duplicate messages being forwarded and thus leads to a significant amount of unnecessary energy consumptions. On the other hand, in gossiping, received messages are only forwarded with some pre-defined probability[4]. By theoretical analysis, a threshold probability exists to cover the whole network with high probability for a given topology and wireless communication loss. Thus by setting the pre-defined probability just above the threshold, a great amount of duplicate messages can be avoided. Nevertheless, in practice, the pre-defined probability is very sensitive to the changes of the network topology and wireless communication loss, which often leads to unsatisfactory reliability for message delivery.

Ideally, if without wireless communication loss, every sensor node needs to receive and forward the broadcast message at most once. Thus though their basic forms are known inefficient, significant efforts have been made toward enhancing the efficiency of the flooding or gossiping, while retaining their robustness in the presence of error-prone transmissions.

B. Different Enhancements

The author of [54] proposed a protocol named LM-PB (Lifetime Maximizing Protocol for Broadcasting) that uses a timing heuristic to reduce redundant message forwardings in the basic flooding as well as to extend the network lifetime. To suppress duplicate forwardings, a node only schedules a forwarding when it receives a broadcast message for the first time. Also a short latency named FDL (Forwarding-node Declaration Latency) is introduced before a node forwards a message, and if a forwarding for the same message is overheard, the node cancels its forwarding to further reduce duplicate forwardings. To extend the network lifetime, for a node u, its FDL is computed based on its residual energy $E_t(u)$, specifically, by the following equation

$$FDL(u) = T \cdot (1 - \frac{E_t(u)}{E_{ref}(u)}) + t_D(u) \ , \tag{7}$$

where T is a timing constant, $t_D(u)$ is the maximum delay related to signal processing, transceiver switching and so forth at the potential forwarding nodes other than u, and E_{ref} is the maximum energy capacity of a battery. As a result, each time that several neighboring nodes receive a broadcast message, only the node with the highest residual energy and thus the shortest FDL will forward the message. Other nodes by overhearing will suppress their own forwardings to save the energy so that the network lifetime is extended.

Smart Gossip [55], on the other hand, extends the basic gossip to minimize forwarding overhead while still keeping reasonable reliability. Different from the basic gossip that uses the same static forwarding probability for all sensor nodes, the authors proposed to dynamically adapt the forwarding probability on each node to its local topology and the originator of the broadcast message. Specifically, based on where the forwarded broadcast message comes from and who is its last forwarder, a node's neighbors are divided into three sets, namely, *parent*, *child* and *sibling*. The neighbors in the parent set are those that the node depends on to receive the first forwarded message; the neighbors in the child set are those that depends on the node to receive the first forwarded message; and the remaining neighbors are in the sibling set. Given an expected network delivery ratio τ, the required per-hop delivery ratio τ_{hop} can be estimated by the equation

$$(\tau_{hop})^{\delta} = \tau \ , \tag{8}$$

where δ is the estimated diameter of the network. Thus for a node with K neighbors in its parent set, the required forwarding probability ($p_{required}$) for each parent neighbor can be estimated using the equation

$$(1 - p_{required})^K < (1 - \tau_{hop}) \ . \tag{9}$$

Each node then collects $p_{required}$ from all its child neighbors and uses the maximum as its own forwarding probability. Also, the three sets and $p_{required}$ on each node are computed periodically based on recent message forwarding history, so as to make the forwarding probability adaptive to network dynamics (e.g., node failure).

A more recent work is RBP (Robust Broadcast Propagation) [56], which extends the flooding-based approach and targets for high reliability broadcast. It lets each node do forwarding when receiving the broadcast message for the first time. Then by overhearing, a node can quickly identify the percentage of its neighbors that have successfully received the message. Based on this percentage and the local density (the number of neighbors), a node determines whether to retransmit the message, where the principle is that for a low density, the message will be retransmitted until a high receiving percentage is achieved, while for a high density, a moderate percentage is enough. To counter wireless loss, explicit ACKs will be sent to nodes that are heard rebroadcasting a message several times. In addition, if a node finds itself highly depending on another node to receive broadcast messages, the link between them is deemed as an important link. The downstream node will then notify its upstream node to increase the number of retransmissions to improve the probabilities of message deliveries.

To enhance reliability one step further, the authors of [57] proposed an approach named Trickle with perfect broadcast reliability (i.e. all sensor nodes receive the broadcast message) for code redistribution and update propagation. To keep codes updated, each sensor node transmits a summary of its code if it has not heard a few other sensor nodes do so. When receiving a code summary from its neighbor, a node compares the received summary with its own. If the neighbor's summary is old, the node then sends its new code to the neighbor. And if the neighbor's summary is newer, the node retransmits its own summary so as to trigger the neighbor to send the new code. Otherwise, a node counts the number of summaries received within one time interval, if the number exceeds a threshold, the node suppresses its own transmission so as to save energy. And to balance energy costs, within each time interval, a node randomly picks its summary transmission time by following a uniform distribution. Moreover, the length of a time interval is set to a lower bound when a summary of new codes is

---

[4] In wired networks such as Internet, gossiping was originally designed to let a received message be forwarded to a randomly selected neighboring node. Due to the broadcast nature of wireless communication, gossiping in WSNs is eventually evolved into the version mentioned above.

Dissemination Mechanism	Basic Approach	Balance Energy Consumption	Topology-Aware	Support Duty-Cycle	Reliability	Delay	Message Cost
LM-PB [54]	Flooding	Yes	No	No	Moderate	Moderate	Very low
Smart Gossip [55]	Gossiping	No	Yes	No	High	Low	Very low
RBP [56]	Flooding	No	Yes	No	Very high	Low	Low
Trickle [57]	Flooding	Yes	No	No	Perfect	Moderate	Low
RBS [58]	Flooding	No	No	Yes	Perfect	Moderate	Low

TABLE III
DIFFERENT CONTROL MESSAGE DISSEMINATION MECHANISMS.

received, so as to accelerate code updates. After that, the length of each next interval will be the double of the current one until it reaches to an upper bound, which further helps to reduce energy costs.

C. Integrated with Duty-Cycle

The above approaches, though are designed with different stress, such as reducing energy consumption or assuring high reliability, all take an implicit assumption that all network nodes are active during the broadcast process (referred to as *all-node-active assumption*). This assumption is valid for wired networks and for many conventional multi-hop wireless networks. It however may fail to capture the uniqueness of the energy-constrained applications in wireless sensor networks. In these applications, sensor nodes are often alternating between *dormant* and *active* states [13][14]; in the former, they go to sleep and thus consume little energy, while in the latter, they actively perform sensing tasks and communications, consuming significantly more energy (e.g., 56 mW for IEEE802.15.4 radio plus 6 to 15 mW for Atmel ATmega 128L micro-controller and possible sensing devices on a MicaZ mote). Define *duty-cycle* as the ratio between active period and the full active/dormant period. A low duty-cycle WSN clearly has a much longer lifetime for operation, but breaks the all-node-active assumption. More importantly, the duty-cycles are often optimized for the given application or deployment, and a broadcast service accommodating the schedules is thus expected for cross-layer optimization of the overall system.

To accommodate low duty-cycle in WSNs, the authors of [58] proposed RBS (Reliable Broadcast Service) to dynamically schedule message forwardings by adapting to its neighbors' active-dormant patterns and forwarding schedules. The core idea is to let a node only issues a message forwarding if it finds otherwise some neighbor may miss the message and can not be contacted until next time the neighbor becomes active. In addition, when a broadcast message is received for the first time, a node also schedules a transmission for the message so that the message can be quickly delivered among its active neighbors. Also, when forwarding a broadcast message, a node piggy-backs those neighbors that it knows have received the message. Then by overhearing, other nodes can quickly know which neighbor has received the message even if some forwarded messages have been missed due to being dormant or wireless loss. To further reduce energy costs, after a forwarding, a node assumes all its active neighbors have received the forwarded message and will not try to forward the same message to these neighbors unless after a timeout, the receipts of the message on these neighbors are still not confirmed by overhearing.

D. Summary

The control message dissemination mechanisms discussed in this section are summarized in Tab. III. Although these works are either based on the flooding or the gossiping, their enhancements cover a broad spectrum. Due to the broadcast nature of the wireless communication, messages forwarded by a sensor node may be received by multiple nodes. The topology information thus can be exploited to avoid duplicate messages being sent to the same node. In addition, on a topology there may be critical positions that other nodes rely on to receive messages. Carefully considering these positions may greatly increase the reliability of the whole dissemination. Besides, to extend the network lifetime, only reducing the total message costs may not always be effective. The energy consumption also needs to be balanced among different nodes so as to avoid some nodes to be over-burdened and out of energy too early. Also, there is an implicit tradeoff among the reliability, delay and message costs, since higher reliability or lower delay may introduce more message costs, and sometimes higher reliability also causes more time for the dissemination process to be finished.

Another observation is that although many mechanisms have been proposed, most of them did not consider the scenario of low duty-cycle WSNs except for RBS [58]. Along this new direction, many efforts are still required. First, theoretical models are expected to be introduced to more clearly understand how duty-cycle and the active-dormant patterns would affect the message dissemination. Also, RBS is proposed to achieve perfect broadcast reliability. This however is not mandatory in some scenarios, where it may be preferred to sacrifice a small portion of reliability so as to cut off more message costs. For such scenarios, a gossiping-based approach may be more favored for the system design. Moreover, in a low duty-cycle WSN, although the topology of active nodes changes frequently, the physical topology containing all nodes is relatively stable. Thus how to apply topology-aware techniques such as those used in [55][56] to message dissemination in low duty-cycle WSNs is also an interesting topic.

VI. CONCLUSION

Wireless sensor networks have been applied to many applications since emerging. And sensor data collection is one of the most important applications among them. In a WSN for sensor data collection, sensed data are continuously collected at all or some of the sensor nodes and forwarded through wireless communications to a central base station for further processing. This makes it different from other applications of WSNs as well as traditional sensor data collection using wired networks. In this paper, we presented an in-depth survey on recent advances in networked wireless sensor data collection. Specifically, we first highlighted the special features of sensor data collection in WSNs, by comparing it with both wired sensor data collection networks and other applications using WSNs. Bearing these features in mind, we discussed issues on using WSNs for sensor data collection, which in general can be broken into the deployment stage, the control message dissemination stage and the data delivery stage.

In the deployment stage, based on whether the coverage requirement is area-coverage or location-coverage, different strategies have been proposed to achieve different levels of coverage and network connectivity while minimizing the required node number or maximizing the network lifetime, according to the physical limits (such as the sensing range and communication range) of the sensor nodes.

In the data delivery stage, different approaches have been proposed to deliver sensing data from sensor nodes to the base station and optimize their own main QoS considerations as well as balance the tradeoffs among other QoS requirements, such as improving throughput while considering rate/congestion control and fairness, balancing energy consumption and latency, or enforcing better reliability with more transmission overheads.

The control message dissemination stage, on the other hand, strives to reliably disseminate control messages over the network with low time and transmission costs, where different mechanisms such as forwarding based on the residual energy of sensor nodes or the network topology information have been used to enhance the basic flooding or gossiping to achieve good balances among the reliability, delay and message costs.

Although these stages have their own issues to address, it has been shown that by considering them jointly, better performance can be achieved. One example is to be aware of the traffic pattern in the data delivery stage while designing the strategy for the deployment stage as discussed in Section III. Other examples include to consider the multi-path data delivery enabled by the deployment stage (with k connectivity) as mentioned in Section IV, and to support duty-cycle in the control message dissemination stage in a way similar to the data delivery stage as investigated in Section V.

In the future, many issues still need to be further explored and possibly considered jointly so as to lead to a more efficient and long-lifetime sensor data collection system. Some of the directions are to consider the special many-to-one traffic pattern in the data delivery stage as well as the one-to-many traffic pattern in the control message dissemination stage; also, the sensing environment in practice may be more complicated than a regular 2-D sensing field, where obstacles and elevation differences may reduce the capacity of wireless communication, resulting in various deployment designs and thus complicated network topologies for data delivery and control message dissemination, and therefore needs to be specifically considered during the performance optimization; low duty-cycle is considered as an effective way to extend the network lifetime of a WSN, yet an interesting topic is to explore how its utilization in networked wireless sensor data collection interacts with other design issues; and another direction is to further optimize the system performance by combining the designs of the deployment, data delivery and control message dissemination stages together.

REFERENCES

[1] M. Tubaishat and S. Madria, "Sensor Networks: an Overview," *IEEE Potentials*, vol. 22, no. 2, pp. 20–23, April/May 2003.

[2] G. Tolle, J. Polastre, R. Szewczyk, D. Culler, N. Turner, K. Tu, S. Burgess, T. Dawson, P. Buonadonna, D. Gay, and W. Hong, "A Macroscope in the Redwoods," in *ACM SenSys*, 2005.

[3] L. Selavo, A. Wood, Q. Cao, T. Sookoor, H. Liu, A. Srinivasan, Y. Wu, W. Kang, J. Stankovic, D. Young, and J. Porter, "LUSTER: Wireless Sensor Network for Environmental Research," in *ACM SenSys*, 2007.

[4] G. Barrenetxea, F. Ingelrest, G. Schaefer, and M. Vetterli, "SensorScope: Out-of-the-Box Environmental Monitoring," in *ACM/IEEE IPSN*, 2008.

[5] G. WernerAllen, K. Lorincz, J. Johnson, J. Lees, and M. Welsh, "Fidelity and Yield in a Volcano Monitoring Sensor Network," in *USENIX OSDI*, 2006.

[6] W.-Z. Song, R. Huang, M. Xu, A. Ma, B. Shirazi, and R. LaHusen, "Air-dropped Sensor Network for Real-time High-fidelity Volcano Monitoring," in *ACM MobiSys*, 2009.

[7] Y. Kim, T. Schmid, Z. M. Charbiwala, J. Friedman, and M. B. Srivastava, "NAWMS: Nonintrusive Autonomous Water Monitoring System," in *ACM SenSys*, 2008.

[8] S. Kim, S. Pakzad, D. Culler, J. Demmel, G. Fenves, S. Glaser, and M. Turon, "Health Monitoring of Civil Infrastructures Using Wireless Sensor Networks," in *ACM/IEEE IPSN*, 2007.

[9] M. Ceriotti, L. Mottola, G. P. Picco, A. L. Murphy, S. Guna, M. Corrà, M. Pozzi, D. Zonta, and P. Zanon, "Monitoring Heritage Buildings with Wireless Sensor Networks: The Torre Aquila Deployment," in *ACM/IEEE IPSN*, 2009.

[10] C. Hartung, R. Han, C. Seielstad, and S. Holbrook, "FireWxNet: A Multi-Tiered Portable Wireless System for Monitoring Weather Conditions in Wildland Fire Environments," in *ACM MobiSys*, 2006.

[11] N. A. Pantazis and D. D. Vergados, "A Survey on Power Control Issues in Wireless Sensor Networks," *IEEE Communications Surveys and Tutorials*, vol. 9, no. 4, pp. 86–107, 2007.

[12] R. Rajagopalan and P. K. Varshney, "Data-Aggregation Techniques in Sensor Networks: A Survey," *IEEE Communications Surveys and Tutorials*, vol. 8, no. 4, pp. 48–63, 2006.

[13] Y. Gu, J. Hwang, T. He, and D. H.-C. Du, "μSense: A Unified Asymmetric Sensing Coverage Architecture for Wireless Sensor Networks," in *IEEE ICDCS*, 2007.

[14] X. Wang, G. Xing, Y. Zhang, C. Lu, R. Pless, and C. Gill, "Integrated Coverage and Connectivity Configuration in Wireless Sensor Networks," in *ACM SenSys*, 2003.

[15] W. B. Heinzelman, A. P. Chandrakasan, and H. Balakrishnan, "An Application-Specific Protocol Architecture for Wireless Microsensor Networks," *IEEE Transactions on Wireless Communications*, vol. 1, no. 4, pp. 660–670, October 2002.

[16] S. Olariu and I. Stojmenovic, "Design Guidelines for Maximizing Lifetime and Avoiding Energy Holes in Sensor Networks with Uniform Distribution and Uniform Reporting," in *IEEE INFOCOM*, 2006.

[17] N. Xu, S. Rangwala, K. K. Chintalapudi, D. Ganesan, A. Broad, R. Govindan, and D. Estrin, "A Wireless Sensor Network For Structural Monitoring," in *ACM SenSys*, 2004.

[18] C. Gui and P. Mohapatra, "Power Conservation and Quality of Surveillance in Target Tracking Sensor Networks," in *ACM MobiCom*, 2004.

[19] C. Intanagonwiwat, R. Govindan, and D. Estrin, "Directed Diffusion: A Scalable and Robust Communication Paradigm for Sensor Networks," in *ACM MobiCom*, 2000.

[20] R. C. Shah, S. Roy, S. Jain, and W. Brunette, "Data MULEs: Modeling a Three-tier Architecture for Sparse Sensor Networks," in *IEEE International Workshop on Sensor Network Protocols and Applications*, 2003.

[21] M. Ma and Y. Yang, "SenCar: An Energy-Efficient Data Gathering Mechanism for Large-Scale Multihop Sensor Networks," *IEEE Transactions on Parallel and Distributed Systems*, vol. 18, no. 10, pp. 1476–1488, October 2007.

[22] S. Meguerdichian, F. Koushanfar, M. Potkonjak, and M. B. Srivastava, "Coverage Problems in Wireless Ad-hoc Sensor Networks," in *IEEE INFOCOM*, 2001.

[23] J. Ko, Y. Ni, H. Zhou, J. Wang, and X. Zhou, "Investigation Concerning Structural Health Monitoring of an Instrumented Cable-Stayed Bridge," *Structure and Infrastructure Engineering*, 2008.

[24] *Structural Health Monitoring for Guangzhou New TV Tower using Sensor Networks*. [Online]. Available: http://www.cse.polyu.edu.hk/benchmark/

[25] R. Iyengar, K. Kar, and S. Banerjee, "Low-coordination Topologies for Redundancy in Sensor Networks," in *ACM MobiHoc*, 2005.

[26] X. Bai, Z. Yun, D. Xuan, T. H. Lai, and W. Jia, "Deploying Four-Connectivity and Full-Coverage Wireless Sensor Networks," in *IEEE INFOCOM*, 2008.

[27] X. Bai, D. Xuan, Z. Yun, T. H. Lai, and W. Jia, "Complete Optimal Deployment Patterns for Full-Coverage and k-Connectivity ($k \leq 6$) Wireless Sensor Networks," in *ACM MobiHoc*, 2008.

[28] H. Zhang and J. Hou, "On Deriving the Upper Bound of α-Lifetime for Large Sensor Networks," in *ACM MobiCom*, 2004.

[29] H. Zhang and J. C. Hou, "Is Deterministic Deployment Worse than Random Deployment for Wireless Sensor Networks?" in *IEEE INFOCOM*, 2006.

[30] G. Xing, X. Wang, Y. Zhang, C. Lu, R. Pless, and C. Gill, "Integrated Coverage and Connectivity Configuration in Wireless Sensor Networks," *ACM Transactions on Sensor Networks*, vol. 1, no. 1, pp. 36–72, 2005.

[31] X. Bai, S. Kumar, D. Xuan, Z. Yun, and T. H. Lai, "Deploying Wireless Sensors to Achieve Both Coverage and Connectivity," in *ACM MobiHoc*, 2006.

[32] K. Xu, H. Hassanein, and G. Takahara, "Relay Node Deployment Strategies in Heterogeneous Wireless Sensor Networks: Multiple-Hop Communication Case," in *IEEE SECON*, 2005.

[33] X. Cheng, D. Z. Du, L. Wang, and B. Xu, "Relay Sensor Placement in Wireless Sensor Networks," *Spinger Wireless Networks*, vol. 14, no. 3, pp. 347–355, 2008.

[34] G. Lin and G. Xue, "Steiner Tree Problem with Minimum Number of Steiner Points and Bounded Edge-Length," *Information Processing Letters*, vol. 69, pp. 53–57, 1999.

[35] J. L. Bredin, E. D. Demaine, M. Hajiaghayi, and D. Rus, "Deploying Sensor Networks with Guaranteed Capacity and Fault Tolerance," in *ACM MobiHoc*, 2005.

[36] A. Kashyap, S. Khuller, and M. Shayman, "Relay Placement for Higher Order Connectivity in Wireless Sensor Networks," in *IEEE INFOCOM*, 2006.

[37] W. Zhang, G. Xue, and S. Misra, "Fault-Tolerant Relay Node Placement in Wireless Sensor Networks: Problems and Algorithms," in *IEEE INFOCOM*, 2007.

[38] F. Wang, D. Wang, and J. Liu, "Traffic-Aware Relay Node Deployment for Data Collection in Wireless Sensor Networks," in *IEEE SECON*, 2009.

[39] G. Xue, T. P. Lillys, and D. E. Dougherty, "Computing the Minimum Cost Pipe Network Interconnecting One Sink and Many Sources," *SIAM Journal of Optimization*, vol. 10, no. 1, pp. 22–42, October 1999.

[40] N. Aitsaadi, N. Achir, K. Boussetta, and G. Pujolle, "Potential Field Approach to Ensure Connectivity and Differentiated Detection in WSN Deployment," in *IEEE International Conference on Communications*, 2009.

[41] ——, "Multi-Objectives WSN Deployment: Quality of Monitoring, Connectivity and Lifetime," in *IEEE International Conference on Communications*, 2010.

[42] G. Lu, B. Krishnamachari, and C. S. Raghavendra, "An Adaptive Energy-Efficient and Low-Latency MAC for Data Gathering in Wireless Sensor Networks," in *IEEE IPDPS*, 2004.

[43] W.-Z. Song, F. Yuan, and R. LaHusen, "Time-Optimum Packet Scheduling for Many-to-One Routing in Wireless Sensor Networks," in *IEEE MASS*, 2006.

[44] G.-S. Ahn, E. Miluzzo, A. T. Campbell, S. G. Hong, and F. Cuomo, "Funneling-MAC: A Localized, Sink-Oriented MAC For Boosting Fidelity in Sensor Networks," in *ACM SenSys*, 2006.

[45] K. S. J. Pister and L. Doherty, "TSMP: Time Synchronized Mesh Protocol," in *IASTED International Symposium on Distributed Sensor Networks (DSN)*, 2008.

[46] L. Paradis and Q. Han, "TIGRA: *Ti*mely Sensor Data Collection Using Distributed *Gra*ph Coloring," in *IEEE PerCom*, 2008.

[47] C. T. Ee and R. Bajcsy, "Congestion Control and Fairness for Many-to-One Routing in Sensor Networks," in *ACM SenSys*, 2004.

[48] N. Burri, P. von Rickenbach, and R. Wattenhofer, "Dozer: Ultra-Low Power Data Gathering in Sensor Networks," in *ACM/IEEE IPSN*, 2007.

[49] S. Lindsey, C. Raghavendra, and K. M. Sivalingam, "Data Gathering Algorithms in Sensor Networks using Energy Metrics," *IEEE Transactions on Parallel and Distributed Systems*, vol. 13, no. 9, pp. 924–935, September 2002.

[50] S. Deering, "Scalable Multicast Routing Protocol," Ph.D. dissertation, Stanford University, 1989.

[51] S. Floyd, V. Jacobson, C. Liu, S. McCanne, and L. Zhang, "A Reliable Multicast Framework for Light-weight Sessions and Application Level Framing," *IEEE/ACM Transactions on Networking*, vol. 5, no. 6, pp. 784–803, 1997.

[52] S.-Y. Ni, Y.-C. Tseng, Y.-S. Chen, and J.-P. Sheu, "The Broadcast Storm Problem in a Mobile Ad Hoc Network," in *ACM MobiCom*, 1999.

[53] K. Akkaya and M. Younis, "A Survey on Routing Protocols for Wireless Sensor Networks," *Ad Hoc Networks*, vol. 3, no. 3, pp. 325–349, May 2005.

[54] X. Guo, "Broadcasting for Network Lifetime Maximization in Wireless Sensor Networks," in *IEEE SECON*, 2004.

[55] P. Kyasanur, R. R. Choudhury, and I. Gupta, "Smart Gossip: An Adaptive Gossip-based Broadcasting Service for Sensor Networks," in *IEEE MASS*, 2006.

[56] F. Stann, J. Heidemann, R. Shroff, and M. Z. Murtaza, "RBP: Robust Broadcast Propagation in Wireless Networks," in *ACM SenSys*, 2006.

[57] P. Levis, N. Patel, D. Culler, and S. Shenker, "Trickle: A Self-Regulating Algorithm for Code Propagation and Maintenance in Wireless Sensor Networks," in *USENIX NSDI*, 2004.

[58] F. Wang and J. Liu, "RBS: A Reliable Broadcast Service for Large-Scale Low Duty-Cycled Wireless Sensor Networks," in *IEEE International Conference on Communications*, 2008.