

# Data Dissemination and Remote Control in Wireless Sensor Networks



Xiaolong Zheng and Yuan He

**Abstract** Wireless Sensor Networks (WSNs) have been applied in a variety of application areas. Most WSN systems, once deployed, are expected to operate for a long period. Besides gathering data to the sink, reliable and efficient delivery of data or control messages from the sink to the sensors are an equally important task. During the lifetime, it is necessary to fix bugs, reconfigure system parameters, and upgrade the software to achieve reliable system performance. However, manually collecting all nodes back and reconfiguring the nodes through serial connections with computer is infeasible because it is labor-intensive and inconvenient. Data dissemination and remote control over a multi-hop network are desired to facilitate such tasks. In this book chapter, we are going to present the challenges and research space of data dissemination and remote control in WSNs, review existing approaches, introduce relevant techniques, assess various performance metrics, compare representative methodologies, and discuss the potential directions. We compare and elaborate on the existing approaches in two categories: structure-less approaches and structure-based approaches, classified by whether or not the network structure information is used during the disseminating process. Meanwhile, application of emerging techniques like concurrent transmissions in data dissemination further broadens the design space. We will also introduce the latest progress in those relevant directions.

## 1 Introduction

Wireless Sensor Networks (WSNs) have been attracting extensive attention from both the scientific community and industry [1, 2]. The development of low-cost, low-power, multi-functional sensor devices, boosting the application of WSNs. Real application systems are extensively deployed in a variety of application areas such as

---

X. Zheng · Y. He (✉)

School of Software, Beijing National Research Center for Information Science and Technology, Tsinghua University, Beijing, China  
e-mail: heyuan@mail.tsinghua.edu.cn

structural health protection [3, 4], environment monitoring [5–7], underground coal mine monitoring [8], event detection in targeted monitoring areas [9], and habitat monitoring [10].

To support aforementioned applications, remote control in WSNs is essential. Most of these WSN systems, once deployed, are intended to operate unattended for quite a long period. During their lifetimes, it is inevitable to fix software bugs [11, 12], remotely control system parameters [13, 14], and reprogramming the software [15, 16] in order to achieve reliable system performance. Unfortunately, for a large WSN system, manually collecting and reconfiguring nodes are infeasible because some WSN systems are deployed in areas where it is physically impossible for human beings to access [17]. It is also labor-intensive due to the huge number of nodes. Hence, both the system users and administrators require controlling WSNs remotely to manage the system services and system operation.

Disseminating data and control commands over a multi-hop network are a core building block of remote control in WSNs. As shown in Fig. 1, data dissemination represents delivering interested data (control commands, running parameters, new codes, etc.) from the sink node to all or selected nodes in the network, over multi-hop transmissions. Data dissemination is an inverse of data collection that generates data flows from sensing nodes to the sink node to mainly delivery sensing data, as shown in Fig. 2. Data dissemination and collection need to cooperate in the software-defined wireless networks to collect data and disseminate routing control messages [18–20].

In this book chapter, we discuss the challenges and the design requirements in data dissemination and remote control. We introduce existing methods by dividing them into two categories: structure-less schemes and structure-based schemes. Generally speaking, according to different assumptions about the knowledge of network structure information, existing data dissemination schemes can be classified into these two categories. In structure-based schemes, it is assumed that knowledge of network structure information such as node location or network topology is

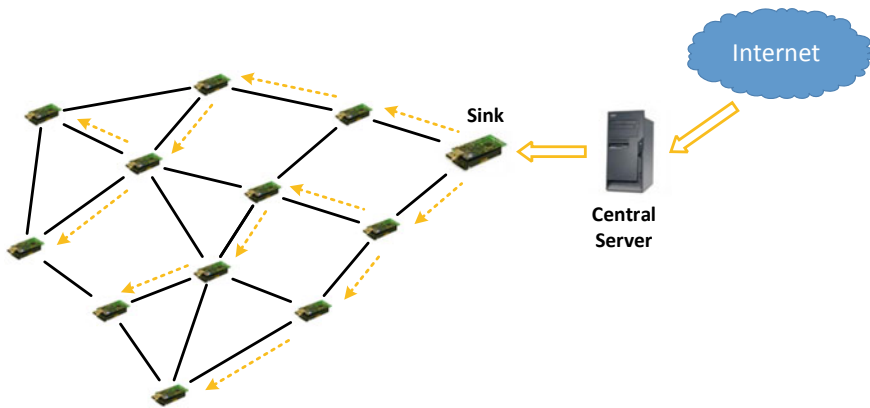


Fig. 1 Data dissemination in WSN

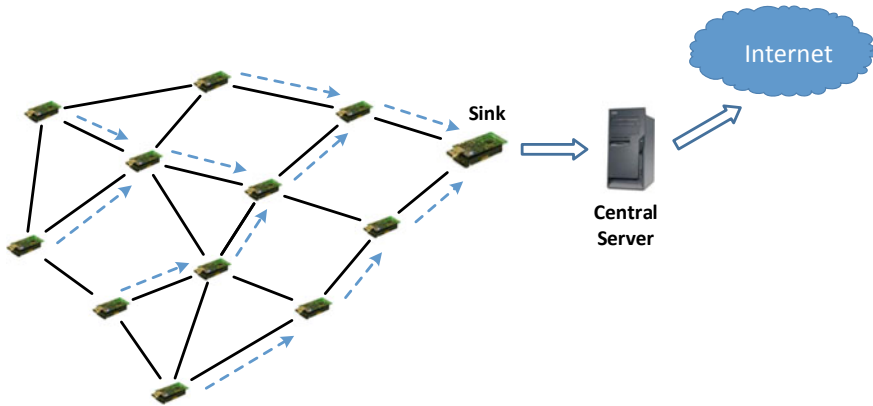


Fig. 2 Data collection in WSN

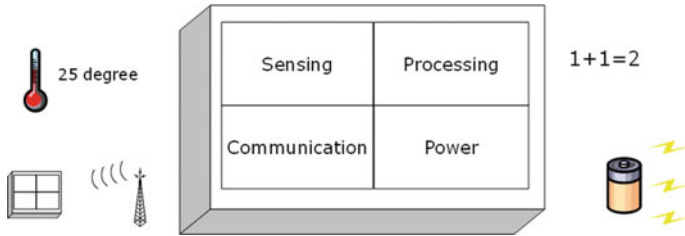
available. Dissemination methods can then take advantage of this knowledge to construct a dedicated structure for efficient dissemination. In structure-less schemes, there is no information of network structure or dedicated structure for dissemination.

For structure-less schemes, according to using negotiation mechanism or not, we further divide them into non-negotiation schemes and negotiation-based schemes. In negotiation mechanisms, the redundant transmissions are under control and the reliability is usually guaranteed. However, negotiation is a double-edged sword. It brings about additional communication overhead and time consumption as well. In non-negotiation schemes, without control message, the dissemination process is relatively quick but providing arbitrarily high reliability is nearly impossible. Besides, broadcast storm problem is easy to occur if non-negotiation schemes do not properly control the broadcast of nodes.

In structure-based schemes, two sub-categories, plain-structure and hierarchy-structure schemes, are introduced. In plain-based schemes, all nodes have the same role in the disseminating process. The structure information is only used for controlling the redundancy but not the forwarding strategy. On the other hand, in hierarchy-structure schemes, the structure information is used to construct a special structure dedicated for data disseminating. The network is usually divided into clusters with a cluster head for each. The cluster heads build up a backbone network to get data preferentially. And then the cluster headers disseminate data to the cluster members in their own clusters in parallel.

In this book chapter, we analyze the merits and demerits of corresponding schemes, along with the trade-offs between different categories. In the existing literature, different categories usually have definite boundaries and no trade-off has been analyzed before. The hybrid schemes, combining non-negotiation and negotiation-based schemes, will be discussed and analyzed in this chapter.

The rest of this chapter is organized as follows. In Sect. 2, we discuss the challenges and the requirements of data dissemination. Then we introduce the structure-less data dissemination scheme in Sect. 3 and structure-based data dissemination scheme in



**Fig. 3** Typical components of a WSN node

Sect. 4. In Sect. 5, we present some related techniques that can be employed in data dissemination. Then, we summarize performance measurement metrics and present the performance comparisons of existing data dissemination schemes in Sect. 6. We further discuss the open issues in Sect. 7 and give the conclusion in Sect. 8.

## 2 Requirements and Challenges

In this section, we first review the features of WSNs in Sect. 2.1. Then we analyze the data dissemination requirements in Sect. 2.2 and summarize the corresponding challenges satisfying the requirements in Sect. 2.3.

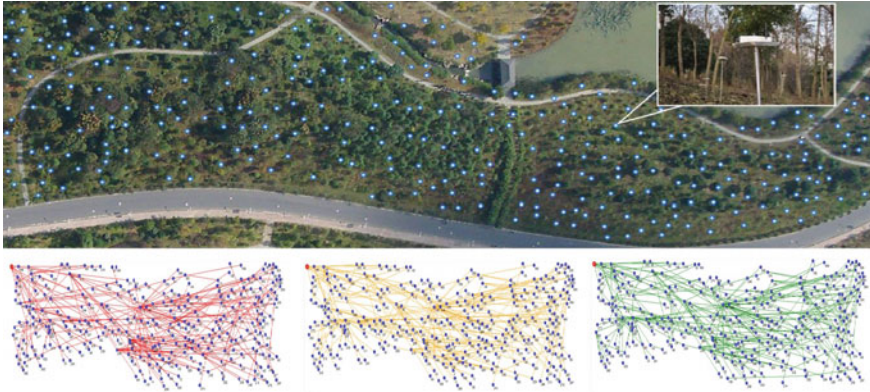
### 2.1 Features of WSNs

In WSN systems, it is common to deploy many sensor nodes in targeted areas which may be physically inaccessible for human beings. For example, for virgin forest monitoring application such as GreenOrbs [5], the deployed environment is not easy to access for system administrators to maintain the network on site.

A typical WSN node consists of four components: power, processing, sensing, and communication, as shown in Fig. 3. The power component is usually using battery when the target environment is outdoor without power supply around. Since the power is limited, other components have to be as low power as possible.

Due to the simple hardware of nodes and application requirements, compared to traditional wireless communication systems, WSNs have the following outstanding features.

- Large scale. Due to applications of WSN usually requiring large deployment area, the amount of sensor nodes can be very large because the communicating range is only dozens of meters.
- Long term. The WSN system is usually intended to work for a long time after deployment. A long lifetime is key to good applicability.



**Fig. 4** Deployed prototype of GreenOrbs and the network topology

- **Self-organized.** WSNs are ad hoc networks which organize and manage the network by themselves. Hence, nodes usually adopt self-repairing and recovery mechanisms to be fault tolerant.
- **Limited resources.** As introduced above, sensor nodes are powered by battery, which means limited energy resources [21]. Besides, data storage and computational capacity are also limited because low-power hardware is adopted.
- **Low power.** In WSNs, sensor nodes usually work in a so-called duty-cycle mode [22–24] to fulfill the long-term lifetime requirement with limited energy resource. In duty-cycle mode, a node periodically wakes up for some tasks and then sleeps for most of the time. In sleeping mode, the consumed power is reduced to the minimum.
- **Dynamic topology.** WSNs are ad hoc networks which maintain the network by themselves according to the quality of communication links. But, communication links are easily affected by environmental changes. Therefore, WSNs are usually dynamic because the environment usually keeps changing, as shown in Fig. 4.

## 2.2 Requirements of Data Dissemination

As introduced in Fig. 1, data dissemination spreads data from sink node(s) to all nodes in network over a multi-hop network. Data can be a new code image for application changes, system commands, or updated system parameters for remote control. To satisfy the Quality of Services (QoS), three requirements of data dissemination in WSNs need to meet.

**Reliability.** Data dissemination generally requires 100% or near 100% reliability. Here 100% reliability has two meanings: (1) all the nodes in the network should get the data; (2) every node receives the whole data block without any hiatus. Reliability

is the most important and essential requirement for data dissemination. Because data dissemination is the building block of reprogramming and remote control, any mismatch of above two aspects can lead to inconsistency and even crash of the whole network.

**Efficiency.** Efficiency requirement has two aspects: time efficiency and energy efficiency. To achieve time efficiency, the dissemination process should finish as soon as possible. During the dissemination process, a large number of data dissemination packets occupy the channel. The normal functional collecting traffic is blocked, and the system is inefficacious during this period. Therefore, this interruption time period should be as short as possible.

Energy efficiency desires minimal energy consumption for the data dissemination process. This is the consequence of limited power resources, the feature of WSNs. The consumed energy consists of read–write of flash, radio transmission, and idle listening of radio. The read–write of flash is inevitable to store data blocks if the data block is large. The major of energy consumption comes from radioactivity which should be carefully controlled to achieve energy efficiency. Restricting the radio-on time during data dissemination is the key to energy efficiency. Energy balance between neighboring nodes can also alleviate the energy drain problem of nodes with large data traffic.

**Scalability.** WSN systems usually have various network size and density. A good data dissemination scheme should adjust to any scale, in terms of the number of nodes and node density. From the performance aspect, if the completion time of dissemination is linearly increasing with network scale, the dissemination scheme is regarded as scalable.

### ***2.3 Challenges of Data Dissemination***

Considering the distinctive features of WSNs, data dissemination faces several challenges.

Limited resources bring limitations to the dissemination process. Dissemination must adjust according to the capacity of node platforms. For example, if the disseminated data block are larger than the size of RAM on sensor node, it is impossible to transmit whole data block at one time. For example, Mica [25], released in 2001, has 4 KB RAM and 128 KB flash; Telos [26], released in 2004, has 10 KB RAM and 48 KB flash. If the size of data block is larger than the RAM size, transferring from RAM to flash is necessary.

As discussed in previous section, to be energy efficient, dissemination should try to minimize the radio-on time. The larger the data are, the more energy is consumed. Hence, except for the obligatory transmission cost of data packets, the radio for other extra use such as control messages and idle listening should be reduced as much as possible.

High reliability is always desired by dissemination but it is not trivial to achieve. As we know, the network is dynamic which means nodes in the network may break away from the network and new nodes can join in the network at any time. The connectivity of network keeps changing all the time. Data dissemination should deal with network dynamics and guarantee the new joined nodes also have the ability to catch up and get the latest data.

Due to the larger number of dissemination packets, it is easy to encounter the so-called broadcast storm problem [27]. Too many redundant transmissions result in serious packet collisions and transmission failures. To make data dissemination reliable and efficient, the broadcast storm problem should be avoided.

The simple reverse uses of routing protocol are not efficient. Routing protocols are designed for data collection. Data collection is usually a many-to-one communication model but data dissemination is one-to-many communication model. In data collection, the data flows are bottom-up because all nodes in the network send sensing data to a sink. But in data dissemination, all nodes receive new data from the sink. The generated data flows are top-down and one-to-many. General routing protocols fail to disseminate data efficiently. Dedicated dissemination scheme is necessary to accomplish high-reliable, high-efficient data dissemination. Many protocols tailored to data dissemination are proposed, as introduced in the following sections. Challenges of data dissemination in WSNs are also discussed in a survey [28].

### 3 Structure-Less Data Dissemination Schemes

We call the dissemination methods without dedicated dissemination structure as structure-less data dissemination schemes. The topology information of underlying network is not required for structure-less data dissemination schemes. As a compromise, local optimal strategy is adopted to greedily approximate global optimal solution. The major disadvantage of these schemes is that they are not able to achieve a global optimal solution. The advantage is that no additional overhead costed by getting global topology information and constructing a dedicated structure. Especially in the dynamic networks, frequent topology changes cause significant additional overhead. We further divide structure-less schemes into two categories, non-negotiation schemes and negotiation-based schemes, according to the use of negotiation strategy in a particular scheme.

#### 3.1 Non-negotiation Schemes

The most representative non-negotiation scheme is *Classic flooding*. In classic flooding, the sink node starts the dissemination. Data is broadcasted by sink to all neighbors. Upon receiving a piece of data, the node will check the data are stored before or not. If not, the receiving node will store the data and then checks whether it has

already forwarded the data to its own neighbors. If not, it forwards a copy of the data to its neighbors by broadcast. Otherwise, it remains silent. Classic flooding maintains quite small amount of protocol state and disseminates data quickly in a network. It is simple yet effective.

However, the deficiencies of classic flooding lead to its inadequacy as a protocol for WSNs. First, classic flooding is unreliable. In this approach, there is no automatic repeat request (ARQ) scheme adopted and hence no reliability guarantees. ARQ is an error correction method for data transmission that uses acknowledgments (messages sent by the receiver indicating that it has correctly received a data frame or packet) and time-outs (specified periods of time allowed to elapse before an acknowledgment is to be received) to achieve reliable data transmission over an unreliable service. It is obvious that the fraction of receiving data is decreasing exponentially with the increase in the hop count. However, high reliability is a desired characteristic and a basic requirement. As a result, classic flooding has to repeatedly broadcast data many times to provide a high reliability, resulting in less efficiency.

Second, classic flooding is easy to cause the so-called broadcast storm problem [27]. In classic flooding, no matter the neighbors have already received all data or not, a node receiving new data always rebroadcasts the new data. A node may hear several senders in its transmission ranges due to the broadcast nature of wireless communication. Too many rebroadcasts then lead to redundancy and collision. What's worse, if ARQ is employed for the reliability, the broadcast storm problem is more serious because all receivers send the ACK back immediately after receiving the broadcast packets. Therefore, if not properly used, classic flooding will result in serious redundancy and collision. In [27], the impacts and severity of broadcast storm are analyzed. The results show that the broadcast storm problem is aggravated by high density and large scale.

To conquer the broadcast storm problem, the authors in [27] proposed five revised flooding schemes: probabilistic, counter-based, distance-based, location-based, and cluster-based flooding. Among these schemes, probabilistic and counter-based flooding are two lightweight schemes that only slightly modify classic flooding. Distance-based and location-based schemes need the geography information to reduce the redundancy. Cluster-based scheme is a structure-based scheme, which will be introduced in the following section.

All these revised schemes designed to alleviate the broadcast storm problem are using two ways: (1) reducing broadcast redundancy; (2) differentiating stagger the broadcast time. First of all, all these schemes differentiate the rebroadcast timing by inserting a random back-off before the rebroadcast. Because the radio following IEEE 802.15.4 is simple without techniques like rate adaption in WLANs [29], the proposed methods just drop the broadcast to reduce the interference. Different schemes follow different dropping rules.

*Probabilistic scheme* is a straightforward scheme that reduces the redundant rebroadcasts by assigning a rebroadcast probability. On receiving a fraction of data for the first time, a node will rebroadcast it with the probability  $P$ . Clearly, when  $P = 1$ , probabilistic scheme is equivalent to classic flooding. But when  $P < 1$ , probabilistic scheme can avoid unnecessary rebroadcasts by adjusting the rebroadcast probability.



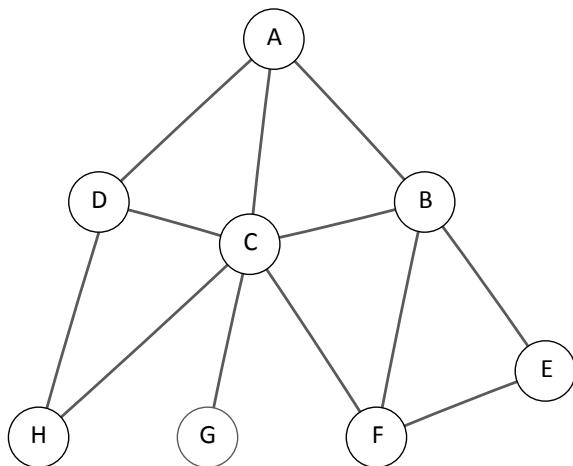
*Counter-based scheme* is also a scheme that tries to reduce the redundancy by suppressing rebroadcast. It is a density-aware method that leverage the density to adjust the rebroadcast behavior. When a node has a new data packet to transmit in the queue, it will overhear the channel status. If more than  $C$  packets identical to the packet in queue are overheard during the waiting period, the node will drop its own transmission.

All the parameters such as  $P$  and  $C$  are preset empirically. When a dynamic network is considered, the fixed preset parameters will be inappropriate. To deal with dynamic network, the authors in [30] propose the revised versions of probabilistic and counter-based flooding schemes. Instead of using the fixed preset parameters as previous work, the proposed schemes allow nodes to choose parameters based on local information. Take the network with topology shown in Fig. 5 as an example. Node B and C should have different rebroadcasting probabilities due to the different local network features. According to [30], the rebroadcasting probability of C should be smaller than the rebroadcasting probability of B. This is because that C's local network and its rebroadcast may bring less benefit and more harm to the whole process.

*Gossip* [31] is proposed to alleviate the storm problem by randomization. Actually, it is also a probabilistic scheme similar to probabilistic flooding [27]. But Gossip is an adaptive gossip protocol.

A node with Gossip will decide its gossip probability based on the number of its neighbors. The probability of a node is in inverse proportion to the number of its neighbors. We regard this scheme as the *Adaptive Neighbor* method. In this scheme, it only considers the transmission efficiency by assigning the probabilities adaptively. However, it ignores the special characteristics of network topology, network connectivity. Again, take the network with topology shown in Fig. 5 as an example. Suppose node A is the sink node who initiates the data dissemination. Based on the network connectivity, we can easily find that node G can only get data from C. Therefore, to

**Fig. 5** An exemplified network topology



reliably disseminate data to whole network, node C must rebroadcast every packet to its child G. However, based on the strategy of *Adaptive Neighbor*, node C will choose a small rebroadcasting probability due to the large number of neighbors.

*Trickle* was proposed in [32] as a version management method that manages how and when the update of codes is performed. In *Trickle*, special data summaries that contain the version of data on the nodes are periodically broadcasted. The data summaries are used to maintain consistency in local areas. If a node receives a data summary with newer data version, it will request the latest data to keep itself consistent to other nodes with new data. *Trickle* adopts an exponential back-off timer to reduce the additional overhead caused by unnecessary summary exchanges. If a local network is consistent, then no data exchange is necessary. Then the interval between two data summaries can be enlarged to reduce overhead. Otherwise, the period is quite short to quickly discover new data and efficiently turn into data dissemination. *Trickle* adopts a suppression mechanism to reduce the unnecessary data summaries. A node will drop its own broadcast when an identical data summary is heard. Besides, *Adaptive Neighbor* scheme is inherited in this work. The rebroadcast probability is also adjusted based on the number of duplicated overhearing of the last message. Consider the topology in Fig. 5, again the deficiency is found. Because node C will overhear duplicate summaries from node A, B, and D, Node C will drop many broadcasts which will cause node G has less opportunities to get the new data.

*Smart Gossip* [33] takes underlying network topology into consideration to solve the above problem. *Smart Gossip* adjusts the rebroadcast probabilities according to the underlying network topology. The reason for the improper decisions in the above two schemes is that a node chooses its rebroadcasting probability independently. Hence, in *Smart Gossip*, the node dependency is defined and used for rebroadcasting probability adjustment. *Smart Gossip* defines node X “depends on” Y as the relationship between X and Y if Y gets data from X. A stronger dependency means the probability that Y is able to get data from other nodes except X is lower. In Fig. 5, we can find node G depends on node C totally. Hence, in *Smart Gossip*, C can learn the knowledge that G depends on it totally. After knowing the dependency, node C will increase the rebroadcasting probability up to 1. Despite of making adjustments based on dependency reflected by topology information, *Smart Gossip* is still divided into structure-less schemes because it only uses local topology which can be obtained in a decentralized manner.

### 3.2 Negotiation-Based Schemes

Negotiation is first proposed in *SPIN* [34, 35]. In negotiation methods, nodes in local areas negotiate with each other about who should be the forwarder and what is to be transmitted before the neighboring nodes really begin transmitting data. The so-called negotiation strategy is designed to conquer the broadcast storm problem and to obtain high reliability. Through negotiation about missing data, which part of the data is still needed to retransmit can be known by the senders. Then only useful

information will be transferred. Besides, the negotiation messages can act as NACKs that request the missing data. Hence, the reliability is guaranteed. Due to the reliability guarantee, it is more common use negotiation-based schemes to accomplish remote control than non-negotiation schemes. On the other hand, the negotiation about who should act as forwarder tries to keep one and only one node transmits the data in a local area. Thanks to the negotiation, the number of redundant data transmissions is dramatically reduced and the broadcast storm problem is avoided.

Generally speaking, negotiation uses three types of messages to do the dissemination through a three-handshake message exchange.

- ADV, advertisement messages. ADV messages specify the data that a sender wants to share in the form of meta-data.
- REQ, request for data. Nodes will send back requests to specify which packets are wanted, after they receiving the ADV.
- DATA, the requested data packets. The source node packs the requested packets and broadcasts it in the type of DATA.

Sensor Protocols for Information via Negotiation (*SPIN*) is a family of negotiation-based adaptive data dissemination protocols. The *SPIN* protocols share two common basic ideas. The first one is information sharing to avoid redundant transmissions. Nodes share the information about which parts of the data they already have and which parts are still needed. However, exchanging real data may be a costly network operation because the size of disseminated data are usually large. However, exchanging the summaries about disseminating data can reduce the cost. Meta-data is then proposed to succinctly and completely describe the latest data. Then the original large data can be identified by the meta-data with small size. It means the meta-data of the distinguishable data will still be distinguishable. On the other hand, the indistinguishable data will result in identical meta-data. Second, energy balancing is taken into consideration in *SPIN*. To prolong the lifetime of the system, the nodes monitor their own energy resources and adjust the dissemination behaviors correspondingly.

*SPIN* is the pioneer work that designs negotiation strategy in WSNs. The follow-up work usually adopts the similar negotiation strategy. Each node with *SPIN* broadcasts ADV messages to announce which data they have. If an ADV message is received, the receiver will check its own meta-data and compare it with the meta-data in the ADV message. If the meta-data are not matched, the receiver knows that the ADV sender possesses data with a newer version. The receiver then responds the sender by sending a REQ message that specifies which parts of the data is wanted by the receiver. Once receiving REQ messages, the sender will send out required data encapsulated in DATA messages.

Figure 6 shows an example of the negotiation procedure in *SPIN*. Node A initiates the dissemination. It broadcasts ADV messages to announce its possession of new data (1). Upon receiving ADV messages, node B, C, and D send out the REQ message which contains the information of wanted data parts (2). Node A combines all REQ messages and integrates the wanted data into next transmitted data block. And then, node A broadcasts DATA messages that contain all the data requested by the neighbors (3). Taking the lossy link into account, the DATA packets can get lose due

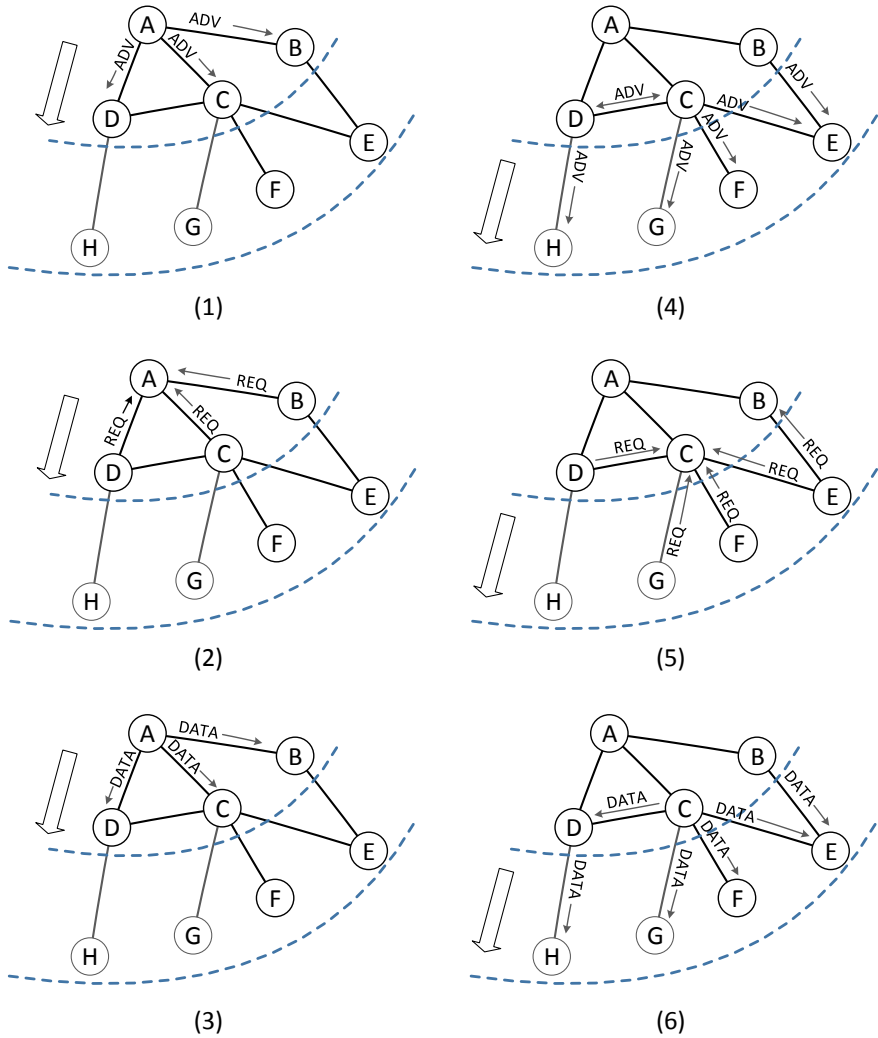


Fig. 6 Negotiation process in SPIN

to the poor link quality. The procedure (2) and procedure (3) are repeated until all the required DATA packets are received or reaching the maximum number of retries. If a node does not get the whole data after the maximum retries, it stops sending REQ anyway to avoid continuously using the poor links. It will wait for the next ADV message to start a new round negotiation and get what it needs from other senders. The nodes can act as forwarders after receiving all the requested data packets. They will broadcast ADV messages to announce their availability as forwarders (4). Then the neighbors of forwarders can request the data by sending REQ messages back to

the forwarders, e.g., node C, after hearing C's ADV message (5). Then by the way same to the one in (2) and (3), node C will broadcast the request data to its neighbors (6). By this hop-by-hop forwarding, data dissemination is accomplished.

In [36], the authors notice that multiple potential forwarders will have channel contention during the forwarding process. All the receivers that have new data can sever as forwarders, which one should be the forwarder in this local areas? If too many forwarders are broadcasting the data at the same time, packet collision between these forwarders will be serious. The dissemination efficiency will be hurt. To deal with above problem, the authors propose *Multi-hop Over-the-Air Programming (MOAP)*. MOAP selects only a small subset of the nodes in a neighborhood to act as forwarders, reducing the unnecessary traffic. Retransmissions are done through unicast to reduce the traffic further. The repair mechanism is a simple sliding window based retransmitting method.

*Deluge* [37] is a reliable data dissemination protocol built on *Trickle* (see Sect. 3.1) to accomplish version control. Similar to *SPIN*, Deluge also adopts a negotiation strategy. Deluge supports bulk data dissemination with a three-handshake negotiation strategy. Deluge can guarantee the reliability by the NACK messages. Data requester should specify the needed packets in the REQ messages. Then the forwarders only retransmit the missing packets to reduce unnecessary transmission cost. Deluge leverages the segmentation and pipelining mechanism which MOAP does not use. We will introduce segmentation and pipelining mechanism in Sect. 5.1. Due to these methods, Deluge is able to exploit the spatial multiplexing to speed up the dissemination process by concurrent transmission without collision. The reliability is guaranteed by hop-by-hop error recovery.

*MNP* [38] shares similar ideas with Deluge. MNP adopts the same negotiation framework as Deluge. The key difference is that a sender selection mechanism is proposed in MNP to reduce the probability of selecting inappropriate forwarder in Deluge. Recall that in Deluge, a candidate is randomly picked out among the neighbors as the forwarder, if there is more than one candidate senders in a local area. However, the number of forwarders will not be bounded and is still large enough to have the forwarder contention problem. What's more, if the randomly selected forwarder has poor link qualities to its neighbors, the completion time will be prolonged. MNP tries to solve this problem by proposing a greedy sender selection algorithm. In MNP, the candidate forwarder with largest expected *impact* will be greedily selected as the next forwarder in a local area. *Impact* is defined as the expected number of neighbors that can get data from a node if they select the node as the forwarder.

*ECD* [39] is an efficient code dissemination protocol that considers the unreliable wireless links. By leveraging one-hop neighbor link quality information, ECD tries to accurately measure the *impact* of a sender. In MNP, *impact* is used as the sender selection metric. However, the *impact* of a node only considers the number of neighbors that need data from the node, without any consideration of the link quality. MNP can perform well in the networks where the links are reliable or near

reliable. But when the links are lossy, the measurement of *impact* is not accurate. Different from MNP, ECD calculates the *impact* with link quality as follows.

$$Impact(i) = \sum_{k \in N(i)} 1 \cdot p(i, k) \quad (1)$$

where  $N(i)$  is the set of the uncovered neighbors of node  $i$ ,  $p(i, k)$  is the link quality from  $i$  to  $k$ . From Eq. (1), we can see the *impact* of node  $i$  in ECD actually reflects the expected number of packets that the uncovered neighbors can receive if node  $i$  is selected as the forwarder.

*ReMo* [40] is a reliable reprogramming protocol. It is tailored to the mobile WSNs. In the mobile WSNs, the location of a sensor node is changing and uncertain. The mobility leads to a dynamic network condition such as varying link qualities. This intrinsic features of mobile WSNs bring challenges to the dissemination protocol design. First, the uncertain locations make it hard to get neighbor information for the potential senders. Second, sender selection algorithms do not work well because some information is not easy to get. For example, the Packet Reception Ratio (PRR) information in ECD is not available in mobile WSNs because the mobility causes continuous changes of the network condition.

To overcome above challenges, ReMo is proposed. Nodes with ReMo learn the neighbors' information such as link qualities and relative distances by measuring the link quality indicator (LQI) and received signal strength indicator (RSSI) of the received packets. Based on the measurement results, the nodes in local area select the best node for data exchanging, with the consideration of the movement paths. ReMo also adopts the polite gossip advertisements similar to *Trickle* (see Sect. 3.1). Broadcasting an advertisement or not is decided by the network density. Another distinguishable feature of ReMo is disordering reception of data. ReMo allows disordered data and rearrange the order after all data are received because the mobility of nodes makes hop-by-hop reliability guarantee hard.

Besides the researches of traditional data dissemination, methods from other researcher areas can be leveraged to accomplish data dissemination. For example, publish/subscribe systems over WSNs [41, 42] can be used for data dissemination. In a publish/subscribe system, a subscriber sends subscribe messages which contain its interest to an event or a predefined topic, to the publisher. The publishers who receive the subscribe messages will send the publish messages according to the notification of subscribers' interests. More details and recent works can be found in the survey [43]. Such a message model is similar to the negotiation-based message model in data dissemination. In publish/subscribe systems, it is the subscribers (normal nodes in the network) initiating the process to pull data from the publishers (sink nodes). However, in data dissemination, it is the sink nodes initiating the dissemination process to push data to the normal nodes in the network. To leverage the methods in publish/subscribe systems, appropriate adjustments are needed.

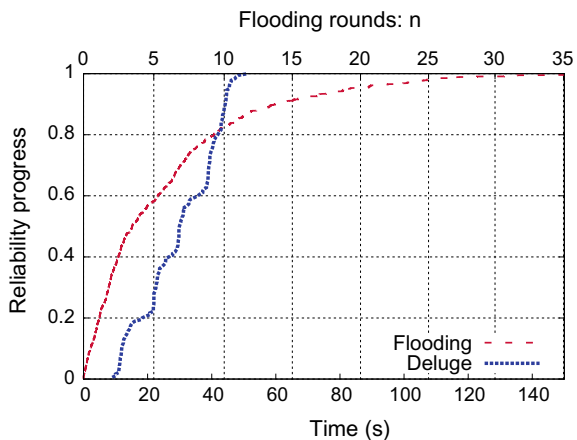
### 3.3 Hybrid Schemes

In the existing literature, different categories have definite boundaries and no trade-off has been analyzed before. Recently, a hybrid scheme [44, 45], combining non-negotiation and negotiation-based schemes, is proposed. Hybrid schemes are proposed to balance the trade-off between efficiency and reliability in the existing framework.

Although the negotiation scheme is essential for guaranteeing a high reliability, it brings additional control overhead. The authors in [45] conduct a motivation experiment on an indoor testbed with 40 TelosB nodes to analyze the performance of non-negotiation and negotiation-based schemes. Probabilistic flooding is used as the representative of non-negotiation schemes, and Deluge (see Sect. 3.2) is the representative of negotiation-based schemes. The network topology is grid, and the distance between two adjacent nodes on the grid is 20 cm. The transmission power is set to the minimum for obtaining the multi-hop dissemination. The rebroadcasting probability in probabilistic flooding is set 0.9. The disseminating data size is 5 KB, packaging into  $48 * 5$  packets. One flooding round is defined as the process that the sink node finishes broadcasting all the packets once. Reliability process is observed during the experiments. The network reliability is defined as the average of the ratio that a node number of unique received packets to the total number of necessary packets.

We cite the experiment results, Fig. 7 from [45], to analyze the trade-off between efficiency and reliability. Figure 7 shows the time-reliability curve for the two classes of schemes. From the results, we can find following observations: (1) for the small number of flooding rounds  $n$ , probabilistic flooding cannot achieve a high reliability. Hence, negotiation is a necessary for 100% reliability. (2) However, for a certain level low reliability, e.g., <30%, probabilistic flooding (with  $n = 1, 3$ ) has a quite much shorter completion time, compared to Deluge. (3) For a high reliability, e.g., >80%, flooding (with  $n = 15$ ) has a larger completion time than Deluge.

**Fig. 7** Reliability progress of flooding and Deluge [45]



We summarize and analyze the observations as follows. (1) For a certain reliability, probabilistic flooding often has a much shorter dissemination time because no control message is involved to defer the data transmissions. (2) But for a higher reliability, probabilistic flooding has to increase the flooding times  $n$  to achieve, bringing inefficiency because the blind flooding without feedbacks tends to cause a large amount of redundancy. In contrast, negotiation can clearly specify the missing packets by explicit requests. The use of negotiation will effectively reduce the redundancy.

There is clear trade-off between negotiation-based schemes and non-negotiation schemes. How to balance the trade-off to get improved performance is the key point. In [44, 45], the authors propose *SurF*, a selective negotiation method, to control the transition time between two schemes to fully explore the benefit of flooding and then change to negotiation-based methods for reliability. For example, one simple hybrid scheme is flooding  $n$  times and then disseminating the remaining data by negotiation-based schemes. The key challenge is to determine when and how nodes transit from one scheme to another one. A bad transition point may result in ruin of the efforts and even bring harm to the dissemination process. For example, if the node turns to negotiation-based schemes after too many rounds of flooding, then the completion time can be longer than the completion time of using negotiation-based schemes only. In contrast, a smaller flooding times  $n$  may result in the insufficient utilization of fast dissemination. The authors model the time-reliability relationships of these two schemes [44, 45]. Based on the time-reliability model, the reliability progresses of these two schemes in the unit time could be modeled and calculated. Then the scheme that can achieve more reliability progress in the next unit time will be adopted.

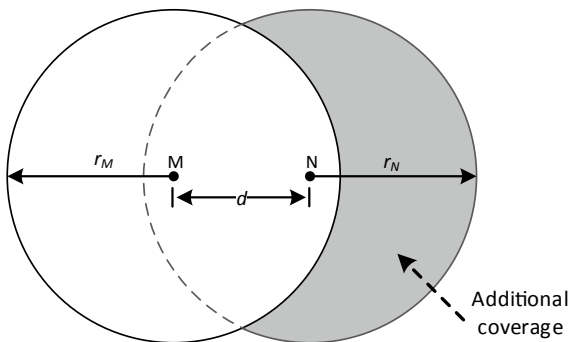
## 4 Structure-Based Data Dissemination Schemes

Different from the structure-less schemes, structure-based schemes take advantages of the network topology and node location information for data dissemination. Even some structure-less schemes use local neighbor information, we do not regard them as structure-based methods because the neighbor information is not enforced to make the methods work. Without the prior topology information, the structure-less schemes can also work. The neighbor information is also very lightweight to learn the local information through the disseminating process. On the other hand, structure-based schemes must have a prior knowledge of the topology information, which is usually the global topology.

We further divide structure-based schemes into two sub-categories: plain and hierarchy-structure-based schemes. In plain-structure schemes, status of all nodes are equivalent in disseminating process, forming a plain structure. In plain-structure schemes, topology information is usually used to help speed the dissemination up or improve the energy efficiency. While in the hierarchy-structure schemes, nodes have different roles in the dissemination process. Normally, nodes will be divided into



**Fig. 8** Additional coverage [27]



clusters with a cluster head for each. Then the selected cluster heads form a backbone network. The data will be disseminated preferentially among the backbone network. And then the cluster heads can disseminate data in their own clusters concurrently to shorten the disseminate completion time.

### 4.1 Plain-Structure Schemes

Two structure-based schemes are proposed in [27] to conquer the broadcast storm problem, with the help of the location information. The first one is called *distance-based scheme*. In distance-based scheme, rebroadcasting a packet or not depends on the distance between neighbors. The benefit of an rebroadcast is defined as the additional coverage the rebroadcast can bring. The additional coverage of a broadcast is defined as the size of new covered area due to this broadcast [27]. Note that even though the mobile sensors can also increase coverage [46] or help the data collection [47], we consider the static network in the following discussion. Figure 8 is an example to illustrate additional coverage. The additional coverage of node N is shown as the gray area. Suppose the distance between M and N as  $d$  where M is a node that just broadcast the same data packet. Then if  $d$  is too small, the additional coverage of N can provide will be little. The additional coverage will become larger when  $d$  is getting larger. The relationship between  $d$  and the additional coverage area  $S_{add}$  of node N is:

$$S_{add} = \pi r^2 - 4 \int_{d/2}^r \sqrt{r^2 - x^2} dx \geq S_T \tag{2}$$

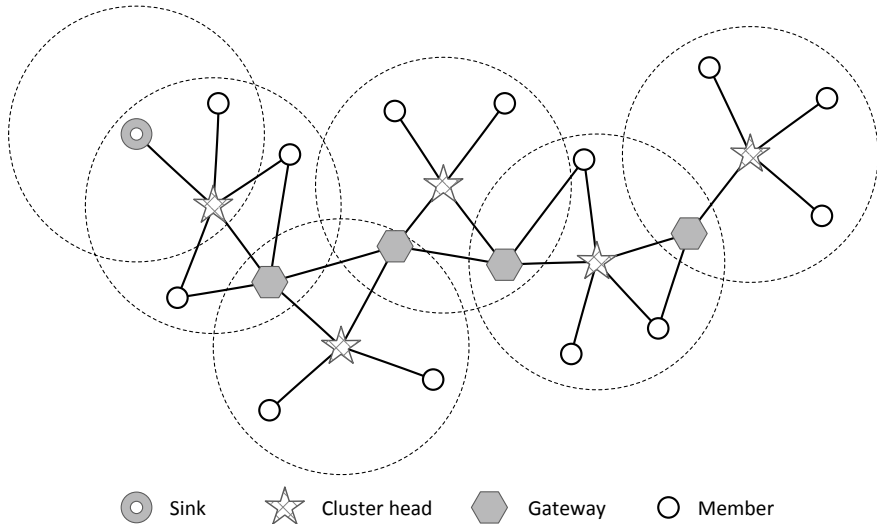
where  $r_M$  and  $r_N$  are the communication radius of node M and N. Here, we simply consider heterogenous network that node M and N have the same radio component. Then the communication radius of node M and N is same,  $r_M = r_N = r$ . From the equation, we can find that if  $d$  is larger than some threshold  $D$ , then the additional coverage will be larger than some threshold. Hence, we can preset a threshold of

additional coverage which is beneficial enough for a broadcast. Then the threshold of  $d$  can be calculated and used for the judgment whether a rebroadcast should be done or not.

The second scheme is *location-based scheme*. This scheme also follows the principle that covering more additional coverage. Note that distance-based scheme has only distance information and makes the judgment of broadcast by distance threshold. When a node has more than one neighbor, distance-based scheme has to use the minimum distance to calculate  $S_{add}$  by Eq. 2 because there is no direction information available. But location-based scheme has the exact locations of nodes provided by powerful hardwares such as global positioning system (GPS). Therefore, nodes with location-based scheme can accurately calculate the  $S_{add}$  instead of estimating by distance  $d$ . In *location-based scheme*, the distance and direction information is available to calculate  $S_{add}$ , taking all the neighbors into account. Then if the calculated  $S_{add}$  is larger than the preset threshold, the node will broadcast. Otherwise, it drops the packet.

*Infuse* [48] is a reliable data dissemination protocol tailored to time-division multiple access (TDMA) MAC layer designs. Infuse needs localization to obtain the neighbor information of the successors in the east/south direction. In TDMA, the time period is divided into slots. Each node in a local area takes an exclusive slot. The nodes are divided into predecessors and successors based on the location information. The dissemination of Infuse is started by the start-download message from the sink. When a node receives the start-download message, it initializes the flash and notifies current running program to prepare for the upcoming dissemination process. Once receiving new data packets, a node will forward the data packet in its own time slots. During a node's transmitting slots, its successors that turn on the radios will receive the forwarded packets. Though TDMA is collision-free, packets get lost because channel fading and external interference causes the unreliable wireless links. To overcome the transmission errors, *Infuse* also designs an error recovery algorithm. A go-back- $n$  based recovery algorithm is used in Infuse. In the go-back- $n$  based recovery algorithm, suppose we have a window of size  $n$ . Then a node will not send packet  $d_i$  before the ACK of packet  $d_{i-n}$  is hard to guarantee the reliability.

*Freshet* [49] mainly focuses on energy efficiency during data dissemination. The basic design is similar to Deluge and MNP (see Sect. 3.2). What's different is that Freshet leverage network topology information to optimize the energy efficiency. In Freshet, when data transmission is not likely to happen in the neighborhood, a node will aggressively turn off the radio to save energy. The first step of data dissemination of Freshet is flooding the topology information to the whole network from sink. Then each node in the network can have an estimation about the time that data may be transmitted in its neighborhood based on this topology information. Before the estimated time of data arrival, a node just turns off its radio and go to sleep state. Such a strategy can help save energy by reducing the radio-on time when the dissemination is far away. Besides, if multiple data sources are available, it allows nodes to receive pages out of the order instead of hop-by-hop reliability guarantee, to speed up the dissemination.



**Fig. 9** An example of a clustered structure

## 4.2 Hierarchical-Structure Schemes

The authors in [27] also propose *cluster-based scheme* which is a hierarchical-structure scheme. Cluster-based scheme uses the topology information to divide the network into several clusters. The cluster construction algorithm is as follows. At the beginning, each node is a cluster with itself as cluster head. Then different clusters are merged together. Every node that is able to communicate with every other node in the merged cluster is a candidate of cluster head. Then the cluster head is decided by the node ID. The candidate with the minimum ID will be elected as the cluster head. The network mobility is also considered in *cluster-based scheme* [27]. When two mobile cluster heads encounter, the node with smaller ID will win the cluster head position. The other cluster will give up the head role and join into the new cluster.

Figure 9 shows an example of the cluster structure. In this cluster-based scheme, nodes are divided into three classes: cluster head as introduce above, gateway, and member. A member node keeps receiving useful data without any rebroadcast. Gateway nodes are special member nodes, locating in the overlapping region of two or more clusters. Hence, the gateways also rebroadcast packets like cluster heads do to help the dissemination process.

*Sprinkler* [50] is a reliable data dissemination protocol. Sprinkler uses the network topology information to construct a hierarchical structure dedicated to dissemination. Sprinkler also designs energy conserving mechanism. In Sprinkler, every node in the network is either a stationary sender or a neighbor connected to one of the stationary senders. We can find that the structure generated by above principle is equivalent

to the connected dominating set (CDS) in a graph. To save energy, less stationary senders should be used. Hence, the problem in Sprinkler is to get the structure with the minimum number of stationary senders. Such a problem is same to the problem that computing the minimum CDS problem which is NP-hard. The authors in Sprinkler propose a low complexity CDS algorithm for the sensor nodes.

After obtaining this hierarchical structure, Sprinkler performs data dissemination via two phases: streaming phase and recovery phase. In the streaming phase, only the nodes in the CDS broadcast packets. In the recovery phase, the non-CDS nodes that have missing data can send out NACK to ask for the wanted data. The CDS nodes with whole data block will respond to the requests and retransmit the requested data. TDMA is also used in Sprinkler to avoid hidden terminal problem.

*Firecracker* [51] is similar to Sprinkler. It combines on-demand routing and data dissemination together to accomplish rapid dissemination. Firecracker has two phases. In the first phase, sink node sends the data to the most distant nodes in the network, leveraging routing paths. Then the nodes traversed by the data from the backbone network naturally. The backbone nodes in Firecracker are the nodes in each corner or randomly selected. Once these distant nodes get the data, the second phase of Firecracker starts. In the second phase, broadcast-based dissemination is used. Data is disseminated along the paths formed in the first phase, like a string of firecrackers. The nodes in backbone networks will broadcast the data to other nodes concurrently.

*CORD* [52] is a reliable data dissemination protocol dedicated to the bulk data dissemination. Similar to the above three hierarchical-structure schemes, CORD also constructs a CDS dedicated to dissemination and adopts two-phase dissemination. The nodes in CORD are divided into core nodes and other normal nodes. In the first phase, data is only disseminated among the core nodes which is a connected dominating set of the network. Then the core nodes broadcast the data to other nodes concurrently. The dissemination among the core nodes can be implemented by the schemes similar to Deluge and MNP (see Sect. 3.2), which disseminates data by reliable hop-by-hop forwarding. Different from above CDS-based methods, CORD adopts a sleep scheduling algorithm in conjunction with the two-phase dissemination to save the energy.

*OAP-SW* [53] is a reliable dissemination protocol. The small world features are analyzed and leveraged in OAP-SW to improve the performance of data dissemination. The small world features indicate that there exist shortcuts from the sink to the other parts of the network. The idea behind OAP-SW is similar to Firecracker. First, sink quickly deliveries the data to some nodes spreading the whole network. Then all these nodes disseminate the data simultaneously to speed up the process. But in OAP-SW, nodes with more powerful hardware are added to act as the endpoints of the shortcuts, forming a heterogeneous network. Then the placement of endpoints of shortcuts is solved by designing the approximation algorithm of the minimum set cover problem. Then the first layer of the hierarchical structure is composed of the powerful nodes, and the second layer is formed by other nodes.

## 5 Other Techniques Used in Data Dissemination

### 5.1 Segmentation and Pipelining

Most data dissemination protocols (e.g., [37–40, 48, 49, 52–54]), especially the one tailored to bulk data dissemination, usually take advantage of pipelining technique to speed up the dissemination. Since wireless communication is intrinsically broadcasting, transmissions collide with each other within the communication range. But outside the communication range, the concurrent transmissions will not be overheard. Pipelining is proposed to exploit such an opportunity of spatial multiplexing.

Segmentation is proposed to cut a large data object into small pieces for more flexible dissemination. For example, in Deluge [37] (introduced in Sect. 3.2), a data object is divided into pages, each of which contains a fixed number of packets (48 packets per page in Deluge). Segmentation is usually used together with pipelining.

The segmentation is illustrated in Fig. 10. The data object is segmented into  $K$  pages. Each page has  $N$  packets. Instead of forwarding the whole data object which is only possible after completely receiving all the data, a node with segmentation technique can act as a forwarder once it receives one complete page. Then the next data page is disseminated. Segmentation helps reducing the waiting time for whole data object and prevents data staying in a single area for too long. Deluge leverages such a segmentation technique together with pipelining to speed up the dissemination process. Most of existing bulk data dissemination protocols follow the same data segmentation principle as Deluge.

Pipelining is proposed with segmentation to further exploit spatial multiplexing. To fully leverage spatial multiplexing, transmissions in different places should be performed simultaneously as much as possible. But the beneficial simultaneous transmissions should not interfere with each other. It is generally assumed that the

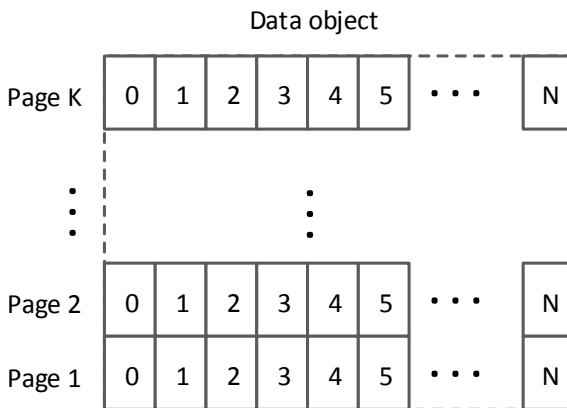


Fig. 10 Segmentation example

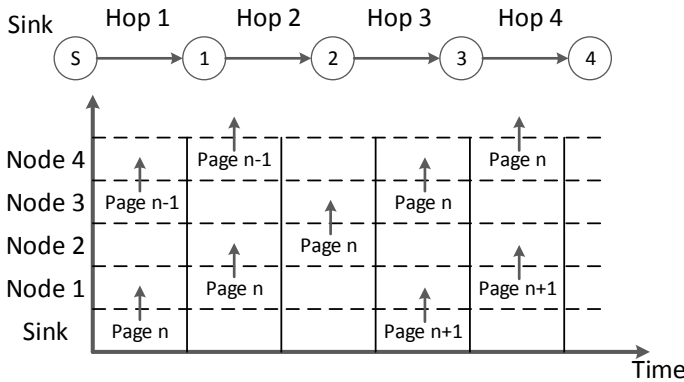


Fig. 11 Pipelining

concurrent transmissions that are three hops away will not collide with current transmission. For example, in Fig. 11, the simultaneous transmissions of sink and node 4 will not collide. They can concurrently transmit two different pages to their own neighbors, speeding up the dissemination.

### 5.2 Coding Technique

Coding is another promising technique used in data dissemination. Traditionally, a receiving vector on the receiver is needed to record the packet receiving status. And a to-transmit vector on the sender side is also needed to record which packets are requested by the receivers and needed to transmit in the next broadcast. This is the root cause why existing methods need NACK or ACK.

But with coding technique, a sender just broadcast the coded packets to the receivers, without the need of knowing that each receiver needs which particular packets. Then partial negotiation procedure can be cut down and the dissemination efficiency can be significantly improved. Generally speaking, the coding techniques used in data dissemination must be rateless because rateless coding technique can make different packets equivalently useful to all the receivers. To guarantee the equivalent status, all the packets are encoded into data blocks that are linearly independent. If the receivers receive enough number of linearly independent encoded packets, they can decode the original data block. When the packets are lost, there is no need for the sender to know which particular packets are lost. In methods with rateless coding, the sender can just keep sending out the encoded packets and the receivers will continue collecting encoded packets until it is enough to decode successfully.

SYNAPSE [55] is a data dissemination protocol leveraging rateless coding. The rateless coding is used in error recovery phase to improve the efficiency. SYNAPSE adopts a low computational complexity called digital fountain codes [56]. The encod-

ing and decoding are just doing the exclusive OR (XOR) operations, which is friendly to the low-power sensor nodes with limited computation resources. Redundant encoded packets will be directly transmitted for the fast error recovery. Negotiation is also used in SYNAPSE. If there is still some receivers who cannot decode the packets, the hybrid ARQ is adopted to do the negotiation. Different from the NACK/ACK in previous negotiation-based schemes that specifies the receiving vector, SYNAPSE only requires the receiver to specify the number of the additional packets it needs to perform decoding.

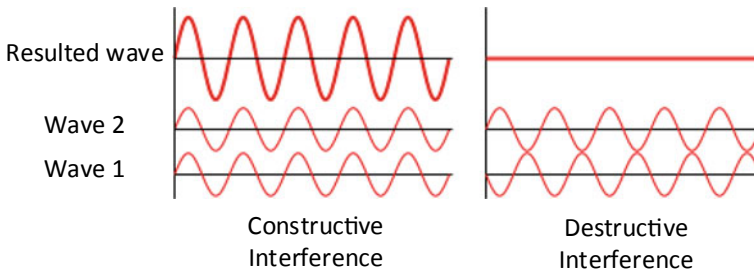
*Rateless Deluge* [57] is another work that leverages coding technique. In Rateless Deluge, two coding-based schemes are proposed: *rateless Deluge* and *ACKless Deluge*. In *rateless Deluge*, a rateless coding technique, random linear coding, is used for disseminating data. The data object will be encoded into  $k$  segments into the  $m$  linear combinations by random linear coding, where  $m > k$ . After encoding, the forwarder broadcasts the encoded packets. Then the receiver that gets any  $k$  messages can recovery the original data image by solving the corresponding system of linear equations.

In *ACKless Deluge*, the second scheme of [57], forward erasure correction (FEC) mechanism is further used to eliminate the need for the control packets. The random linear coding requires the receiver have more than  $k$  messages to decode the original data block. Considering unreliable wireless links, the sender must send out more than  $k$  packets. ACKless Deluge transmits the extra encoded packets in advance to reduce the probability that a receiver does not get enough packets and has to send NACK to request. But the problem is how many extra encoded packets should be transmitted. Too many encoded packets in advance will bring redundancy and waste channel resources unnecessarily. On the other hand, transmitting a small number of encoded packets in advance cannot efficiently avoid time-consuming NACK requests. Hence, ACKless Deluge integrated with FEC tries to provide enough extra encoded packets with little redundancy.

The authors in [58] propose a dissemination method for the duty-cycled WSNs. It saves energy by minimizing the number of transmissions. They propose a XORs encoding decision algorithm to minimize the number of transmissions. The effectiveness of current sleep scheduling on energy saving is analyzed and used for the decision about whether the current sleep scheduling is effective or not.

The authors in [59] provide an analytical upper bound of the completion time for coding-based methods in the WSNs with a general network topology. The results show that the completion time of coding-based dissemination methods is between  $O(N)$  and  $O(N^2)$  where  $N$  is the number of nodes in the network. However, in the proposed completion time model, only the network transmission latency is considered. Encoding time and decoding time are not considered.

The authors in [60, 61] propose a showcase that applying network coding to data dissemination in WSNs. They analyze the effectiveness of coding technique for small values, specifically for a single value dissemination. Traditional works usually focus on bulk data dissemination with coding technique. In [60, 61], CodeDrip is proposed to validate the performance of using network coding for small values. The results



**Fig. 12** Constructive interference and destructive interference

show that CodeDrip is faster, smaller, and sends fewer messages than traditional dissemination protocols for small values such as Drip [13] and DIP [14].

### 5.3 *Constructive Interference*

Recently, there is a trend of exploiting interference instead of avoiding it, such as concurrency [62–66]. Some data dissemination methods that leverage constructive interference are proposed. The proposed methods significantly change the working framework of traditional dissemination methods.

First of all, we introduce constructive interference [67]. As shown in Fig. 12, when two waves with same frequency meet at the same point, the amplitude of resulted wave is the sum of the individual amplitudes. Traditionally, wave 2 is regarded as interference when wave 1 is the interested signal. However, we can find that under constructive interference, wave 1 is amplified when wave 2 is same to wave 1. On the contrary, if wave 2 arrives at phase  $180^\circ$ , the crest of wave 1 meets the trough of wave 2, then destructive interference occurs and the resulted wave is interfered most seriously.

Another physical phenomenon is capture effect which can be also used in WSNs [68–71]. Capture effect is also called co-channel interference tolerance. In capture effect, the receiver is able to correctly receive a strong signal from one sender in spite of the significant interference from other senders.

*Glossy* [72] is a recent work that designs and implements a fast flooding scheme that leverages constructive interference to improve the performance of existing time synchronization [73]. *Glossy* designs a novel flooding architecture for data dissemination in WSNs. *Glossy* exploits CI of IEEE 802.15.4 symbols for the fast network flooding. As introduced in the previous sections, classic flooding is prone to have the broadcast storm problem. One possible approach is to schedule the broadcasts to make them interference free. However, to design such a schedule is a NP-complete problem [74].



Glossy analyzes the underlying reason of CI and intentionally makes it happen beneficially. Glossy requires strictly simultaneous transmissions of the same packet to make them interfere constructively. Then the artificial capture effects help receiver to decode the packet. Glossy disseminates data as follows. The source node initially broadcasts the first packet. Neighbors retransmit the packet right after receiving the packet to meet the conditions of CI which are (1) simultaneous transmissions; (2) same packet. Then the source continues flooding out the data packets after a certain safety time period which is enough for the previous packet floods out the communication range. By this way, Glossy can disseminate the data with a quite high reliability without any control overhead involved in the process.

However, leveraging CI faces several challenges: (1) CI solely cannot provide reliability guarantee; (2) timely error recovery needs careful design because CI keeps the radio too busy to send back ACK or NACK; (3) CI requires more than one reliable transmitters transmit the same packet at the same time. However, the reliable links in real deployment are not always available to provide efficient constructive interference.

A recent work called *Wireless Bus* [75] is proposed. In this work, the network is configured as a bus to do the data transmission without definition of unicast or broadcast. Patching it with the reliability guarantee mechanism is another possible way to do the data dissemination by leveraging CI.

Splash [76] is a dissemination protocol for large data objects in WSNs. Splash combines many techniques to achieve fast and reliable data dissemination. The constructive interference broadcast and multiple-channel pipelining are integrated to eliminate contention overhead among nodes. Without contention, the dissemination completion time can be significantly reduced. Besides, opportunistic overhearing, channel cycling, and XOR coding are also used to ensure the high reliability.

Pando [77, 78] is completely contention-free data dissemination protocol that integrates coding technique and constructive interference. With Pando, data are encoded by Fountain codes. The network is first divided into multiple parallel pipelines, based on constructive interference and channel diversity. And then the sink disseminates the rateless stream of encoded packets along the fast and parallel pipelines.

## 6 Performance Evaluation

The performance of data dissemination methods in WSNs is crucial to the stability and lifetime of the systems. The reason will not be repeated in details because it has been explained in Sect. 2.2. The evaluation is usually done by (1) testbed; (2) simulation. In this section, we introduce the performance metrics and then discuss the advantages and disadvantages of different schemes.

## 6.1 Performance Metrics

To fulfill the requirements of data dissemination, researchers propose several performance metrics to measure the satisfaction degree. We summary the common performance metrics as follows.

**Reliability** is the fundamental requirement as well as the essential performance metric in data dissemination. Measuring reliability usually calculates the ratio of the size of data that nodes in network need to receive, to the size of whole data that all nodes need to receive. For example, if the size of disseminating data is 10KB and 1000 nodes exist in the network, then the size of data that all nodes need to receive is 10,000 KB. If the size of totally received data for all the 1000 nodes in network is 9,800 K according to a protocol, then the reliability of this protocol is 98%. Though sometimes the reliability is not strictly required to be 100%, the 100% reliability is a desired performance. Actually, it is usually required to be a high probability near 100% even if not 100%.

The negotiation-based schemes can achieve a 100% reliability because the negotiation process guarantees that every node gets the latest data eventually in finite time. On the other hand, non-negotiation schemes cannot achieve the 100% reliability in a limited time period because no negotiation helps the nodes with missing data to precisely and actively ask for the wanted data.

**Time efficiency** is crucial for system functionality. Completion time is one of the most important metrics for data dissemination to measure the time efficiency. It is crucial to the overall system performance. The time-efficiency requirement demands dissemination complete as quickly as possible. The completion time of data dissemination is defined as the time period from the first packet that the initial node sends out to the last packet that the last node in the network sends out or receives. In the non-negotiation schemes, the last node finishing broadcasting the data is the end of the dissemination. In the negotiation-based schemes, the last node receiving all the data is the end of the dissemination.

**Energy efficiency** is always the key concern for almost every protocol in WSNs. The less energy a protocol consumes, the longer network lifetime is. Fulfilling the requirements with minimum energy consumption is always desired. On the low-power sensor node platform, radio activities are known as the major energy consumption. Take Mica2 node as example, the energy consumption of some related operations is shown in Table 1 [28]. We can find the current draw when radio works is much larger than the other activities of nodes. Therefore, to reduce the energy consumption to the minimum, the radio-one time should be controlled as short as possible. In negotiation process, the nodes need to turn on their radios for overhearing control messages.

**Memory usage** is another metric to evaluate dissemination performance. Only limited memory resource is available on a sensor node. During the dissemination, memory of certain size is reserved for dissemination data and other necessary information. Generally speaking, segmentation technique can take the memory usage under control. Since the dissemination in segmentation is page by page, the required

**Table 1** Energy consumption on Mica2 platform

Operations	Power consumption ( <i>nAh</i> )
Read a data block from EEPROM	1.261
Write a data block to EEPROM	85.449
Send one packet	20.000
Receive one packet	8.000
Idle listen for 1 ms	1.250

memory size is the size of one-page data and other necessary information such as system parameters, neighbor information. Without the segmentation, the nodes have to store the data into flash storage if the whole data cannot be stored in memory. However, storing data in flash will cause extra delay when a node wants to send out the data because reading from the flash is slower than reading from memory. Besides, the reading and writing on flash also consume additional energy. Hence, for the bulk data dissemination, segmentation is usually adopted.

## 6.2 Performance Comparisons

In this section, we compare different schemes reviewed in this chapter. The detailed performance comparisons of these schemes are omitted because the experiment results reported in different papers cannot compare fairly. The results are obtained under the different experiments or simulation settings vary in different works. Therefore, we just give some general comparisons between different categories as follows.

*Structure-less schemes to structure-based schemes.* Structure-less schemes can be employed in a large-scale system easily because no additional information is needed. They can be very flexible to adapt to the dynamic networks with a good scalability. On the other hand, it is obvious that structure-based schemes are not easy to suit for the dynamic networks. Each change of the network environment can result in a total reconstruction of the dedicated structure which is of heavy overhead. Dedicated structure has side effects. Even though the structure-based schemes reduce control overhead during the dissemination process, they bring about another overhead of maintaining the dedicated structure.

*Non-negotiation schemes to negotiation-based schemes.* Non-negotiation schemes disseminate data quite quickly because no negotiation defers the transmissions of data packets. Besides, non-negotiation schemes do not have the transmission overhead of control messages brought by negotiation. The disadvantage of non-negotiation schemes is that they cannot guarantee the reliability. If high reliability is needed, the ARQ mechanism is necessary. However, the ACK/NACK messages of ARQ mechanism will incur ACK implosion problem and the broadcast storm problem. On the other hand, negotiation-based schemes sacrifice time efficiency for the high

reliability. The negotiation strategy effectively eliminates the redundant transmissions and guarantees the reliability by explicit control messages. The disadvantage of negotiation is time-costly. In recent experiment results reported in [44], the control time incurred mainly by negotiations takes around 70% of the total completion time.

*Plain structures to hierarchical structures.* The plain-structure schemes are more flexible, compared to the hierarchical-structure schemes. There is no cost for acquiring the topology or location information. Some of the plain-structure schemes only need the local structure information. The cost of acquiring local structure information is limited. Another advantage of the plain structure is its convenience to construct and maintain. Compared to the plain structures, hierarchical structures can speed up the dissemination process by dividing the dissemination into two phases and broadcasting concurrently with little interference. Hierarchical-structure schemes are expected to be more efficient in the relatively stationary networks but not in the dynamic networks. If the networks are dynamic, then hierarchical-structure schemes need to reconstruct the disseminating structure once the network changes, which is too costly to maintain.

*Whole image to segmentation.* When data block is in a large size, disseminating the whole data hop by hop is time-consuming. By the segmenting and pipelining, the completion time will be much shorter due to the spatial multiplexing. Except some early works, segmenting and pipelining are widely adopted in the existing works.

Literature [28] and [79] also give some comparisons about the related protocols. In [80], the completion time of negotiation-based schemes is modeled and measured. The dissemination delay in the low-duty-cycle network is analyzed in [81].

## 7 Open Issues

*Security* should be considered in some application systems such as monitoring system for military use. However, the research on security in data dissemination is limited. The traditional encryption algorithms are too complex to operate on nodes. Simple alternative methods are proposed. In [82], the authors change the file system management, reboot mechanism, and bootloader on iMote2 sensor nodes to protect the system from being reprogrammed by an unauthorized third party. The authors in [83] propose a method to achieve confidentiality in the multi-hop data dissemination. Spurious data images from the adversary are prevented by the proposed method. However, these works still require expensive resources and complicated algorithms. To keep real deployed WSNs secure during data dissemination, more effort is needed for designing secure methods more easy and more effective to use.

*Heterogeneous networks* should be considered in the future development of data dissemination. The boom of Internet of things (IoT) brings prosperity of smart devices. More and more IoT and WSN devices will be deployed in our living environment to provide convenience to daily life [84, 85]. It is becoming more and more common many devices from different application systems are deployed in a common area. Even in the same application system, to satisfy different application require-

ments, nodes equipped with various sensors are used. Hence, heterogenous network is an unavoidable network situation.

However, data dissemination among the heterogeneous networks is not well studied. The authors in [86] study the code dissemination problem in heterogeneous WSNs. They formulate the problem as a minimum non-leaf nodes Steiner tree problem. The authors then propose a multicast protocol HSR with an approximate ratio  $\ln |R|$ , where  $R$  is the set of all destinations. Sprinkler [50] can also be applied to heterogeneous networks. However, these methods simply treat the heterogeneous networks as separated networks. When performing data dissemination, they first separate nodes into different sets based on the network nodes belong to. Then the data is disseminated in each separated network.

Recently, researchers propose enabling direct communication among heterogenous devices with incompatible radios [87–93]. With the ability of cross-technology communication, how to disseminate data among heterogenous devices is an quite interesting open issue. More effects are needed to improve the efficiency of data dissemination methods in the heterogeneous networks.

## 8 Conclusion

Data dissemination is a crucial building block for many other system services. During the long lifetime of a WSN system, it is necessary to remotely control the nodes, fix bugs, reconfigure system parameters, and upgrade the software for a better and more reliable system performance. Data dissemination over multi-hops is desired to facilitate such tasks.

Based on using a dedicated structure in the data dissemination or not, we divide existing approaches into two categories: structure-less and structure-based schemes. Structure-less scheme is further divided into negotiation-based and non-negotiation scheme according to that whether or not a negotiation mechanism is used. Structure-based schemes are further divided into plain-structure schemes and hierarchical-structure schemes. We review and compare these schemes in depth. We also elaborate the requirements and challenges of data dissemination in WSNs. We introduce some related and promising techniques such as coding and constructive interference. We also discuss the common performance metrics of data dissemination. We compare different categories and present the corresponding merits and demerits. Finally, we discuss some possible open issues based on the emerging techniques and development trends of WSNs.

Even though plentiful literature exists, there is still space to further improve the data dissemination in WSNs. Besides, some emerging techniques are very promising for changing the working framework of data dissemination. There are still some open issues that need further investigation to design an efficient protocol that can be widely adopted in real deployed systems.

## References

1. Akyildiz, I.F., Su, W., Sankarasubramaniam, Y., Cayirci, E.: Wireless sensor networks: a survey. *Comput. Netw.* **38**(4), 393–422 (2002)
2. Krishnamachari, B.: *Networking Wireless Sensors*. Cambridge University Press (2005)
3. Kim, S., Pakzad, S., Culler, D., Demmel, J., Fenves, G., Glaser, S., Martin, T.: Health monitoring of civil infrastructures using wireless sensor networks. In: *Proceedings of ACM/IEEE IPSN*, pp. 254–263, Apr 2007
4. Xu, N., Rangwala, S., Chintalapudi, K.K., Ganesan, D., Broad, A., Govindan, R., Estrin, R.: A wireless sensor network for structural monitoring. In: *Proceedings of ACM SenSys*, pp. 13–24, Nov 2004
5. Mo, L., He, Y., Liu, Y., Zhao, J., Tang, S., Li, X.Y., Dai, G.: Sustainable sensing in the forest. In: *Proceedings of ACM SenSys, Canopy Closure Estimates with Greenorbs*, pp. 99–112, Nov 2009
6. Mao, X., Miao, X., He, Y., Li, X., Liu, Y.: CitySee: Urban CO<sub>2</sub> monitoring with sensors. In: *Proceedings of IEEE INFOCOM*, pp. 1611–1619 (2012)
7. Liu, Y., Mao, X., He, Y., Liu, K., Gong, W., Wang, J.: CitySee: not only a wireless sensor network. *IEEE Netw.* **27**(5), 42–47 (2013)
8. Li, M., Liu, M.: Underground coal mine monitoring with wireless sensor networks. *ACM Trans. Sensor Netw. (TOSN)* **5**(2), 10 (2009)
9. Li, M., Liu, M., Chen, L.: Nonthreshold-based event detection for 3d environment monitoring in sensor networks. *IEEE Trans. Knowl. Data Eng.* **20**(12), 1699–1711 (2008)
10. Polastre, J., Szewczyk, R., Mainwaring, A., Culler, D., Anderson, J.: Analysis of wireless sensor networks for habitat monitoring. In: *Wireless Sensor Networks*, pp. 399–423 (2004)
11. Ma, Q., Liu, K., Miao, X., Liu, Y.: Sherlock is around: Detecting network failures with local evidence fusion. In: *Proceedings of IEEE INFOCOM*, pp. 792–800, Mar 2012
12. Liu, K., Ma, Q., Zhao, X., Liu, Y.: Self-diagnosis for large scale wireless sensor networks. In: *Proceedings of IEEE INFOCOM*, pp. 1539–1547, Apr 2011
13. Tolle, G., Culler, D.: Design of an application-cooperative management system for wireless sensor networks. In: *Proceedings of EWSN*, pp. 121–132 (2005)
14. Lin, K., Levis, P.: Data discovery and dissemination with dip. In: *Proceedings of IEEE IPSN*, pp. 433–444 (2008)
15. Dong, W., Liu, Y., Chen, C., Gu, L., Wu, X.: Elon: enabling efficient and long-term reprogramming for wireless sensor networks. *ACM Trans. Embed. Comput. Syst.* **13**(4), 17–43 (2014)
16. Mottola, L., Picco, G.P.: Programming wireless sensor networks: fundamental concepts and state of the art. *ACM Comput. Surv.* **43**(3), 19:1–19:51 (2011)
17. Werner-Allen, G., Lorincz, K., Ruiz, M., Marcillo, O., Johnson, J., Lees, J., Welsh, M.: Deploying a wireless sensor network on an active volcano. *Int. Comput.* **10**(2), 18–25 (2006)
18. Guo, J., He, Y., Zheng, X.: PanGu: towards a software-defined architecture for multi-function wireless sensor networks. In: *Proceedings of IEEE ICPADS* (2017)
19. Zeng, D., Li, P., Miyazaki, T., Guo, S., Hu, J., Xiang, Y.: Evolution of software-defined sensor networks. *IEEE Trans. Comput.* **64**(11), 3128–3139 (2015)
20. Cao, C., Luo, L., Gao, Y., Dong, W., Chen, C.: TinySDM: software defined measurement in wireless sensor networks. In: *Proceedings of IEEE/ACM IPSN* (2016)
21. Li, Z., Li, M., Liu, Y.: Towards energy-fairness in asynchronous duty-cycling sensor networks. *ACM Trans. Sens. Netw.* **10**(3), 38 (2014)
22. Zheng, X., Cao, Z., Wang, J., He, Y., Liu, Y.: ZiSense: towards interference resilient duty cycling in wireless sensor networks. In: *Proceedings of ACM SenSys* (2014)
23. Zheng, X., Cao, Z., Wang, J., He, Y., Liu, Y.: Interference resilient duty cycling for wireless sensor networks under co-existing environments. *IEEE Trans. Commun.* **65**(7), 2971–2984 (2017)
24. Cao, Z., He, Y., Ma, Q., Liu, Y.: L2: lazy forwarding in low duty cycle wireless sensor networks. *IEEE/ACM Trans. Netw.* **23**(3), 922–930 (2015)

25. Hill, J.L., Culler, D.E.: Mica a wireless platform for deeply embedded networks. *IEEE Micro*. **22**(6), 12–24 (2002)
26. Polastre, J., Szewczyk, R., Culler, D.: Telos: enabling ultra-low power wireless research. In: *Proceedings of ACM/IEEE IPSN*, pp. 364–369, Apr 2005
27. Ni, S.Y., Tseng, Y.C., Chen, Y.S., Sheu, J.P.: The broadcast storm problem in a mobile ad hoc network. In: *Proceedings of ACM MobiCom*, pp. 153–167, Aug 1999
28. Wang, Q., Zhu, Y., Cheng, L.: Reprogramming wireless sensor networks: challenges and approaches. *IEEE Netw.* **20**(3), 48–55 (2006)
29. Wang, G., Zhang, S., Wu, K., Zhang, Q., Ni, Lionel M.: TiM: fine-grained rate adaptation in WLANs. *IEEE Trans. Mob. Comput.* **15**(3), 748–761 (2016)
30. Tseng, Y.C., Ni, S.Y., Shih, E.Y.: Adaptive approaches to relieving broadcast storms in a wireless multihop mobile ad hoc network. *IEEE Trans. Comput.* **52**(5), 545–557 (2003)
31. Haas, Z.J., Halpern, J.Y., Li, L.: Gossip-based ad hoc routing. In: *Proceedings of IEEE INFOCOM*, pp. 479–491, June 2002
32. Levis, P., Patel, N., Culler, D., Shenker, S.: Trickle: a self-regulating algorithm for code propagation and maintenance in wireless sensor networks. In: *Proceedings of USENIX NSDI*, March 2004
33. Kyasanur, P., Choudhury, R.R., Gupta, I.: Smart gossip: an adaptive gossip-based broadcasting service for sensor networks. In: *Proceedings of IEEE MASS*, pp. 91–100, Oct 2006
34. Heinzelman, W.R., Kulik, J., Balakrishnan, H.: Adaptive protocols for information dissemination in wireless sensor networks. In: *Proceedings of ACM MobiCom*, pp. 174–185, Aug 1999
35. Kulik, J., Heinzelman, W., Balakrishnan, H.: Negotiation-based protocols for disseminating information in wireless sensor networks. *Wireless Netw.* **8**(2/3), 169–185 (2002)
36. Heidemann, J., Stathopoulos, T., Estrin, D.: A remote code update mechanism for wireless sensor networks. Technical report (2003)
37. Hui J.W., Culler, D.: The dynamic behavior of a data dissemination protocol for network programming at scale. In: *Proceedings of ACM SenSys*, pp. 81–94, Nov 2004
38. Kulkarni, S.S., Wang, L.: MNP: Multihop network reprogramming service for sensor networks. In: *Proceedings of IEEE ICDCS*, pp. 7–16, June 2005
39. Dong, W., Liu, Y., Wang, C., Liu, X., Chen, C., Bu, J.: Link quality aware code dissemination in wireless sensor networks. In: *Proceedings of IEEE ICNP*, pp. 89–98, Oct 2011
40. De, P., Liu, Y., Das, S.K.: Remo: an energy efficient reprogramming protocol for mobile sensor networks. In: *Proceedings of IEEE PerCom*, pp. 60–69, March 2008
41. Sarkar, R., Zhu, X., Gao, J.: Double rulings for information brokerage in sensor networks. *IEEE/ACM Trans. Netw.* **17**(6), 1902–1915 (2009)
42. Kündig, S., Leone, P., Rolim, J.: A distributed algorithm using path dissemination for publish-subscribe communication patterns. In: *Proceedings of the 14th ACM International Symposium on Mobility Management and Wireless Access*, pp. 35–42 (2016)
43. Sheltami, T.R., Al-Roubaiey, A.A., Mahmoud, A.S.H.: A survey on developing publish/subscribe middleware over wireless sensor/actuator networks. *Wireless Netw.* **22**(6), 2049–2070 (2016)
44. Zheng, X., Wang, J., Dong, W., He, Y., Liu, Y.: Survival of the fittest: data dissemination with selective negotiation in wireless sensor networks. In: *Proceedings of IEEE MASS*, pp. 443–451, Oct 2013
45. Zheng, X., Wang, J., Dong, W., He, Y., Liu, Y.: Bulk data dissemination in wireless sensor networks: analysis, implications and improvement. *IEEE Trans. Comput.* **65**(5), 1428–1439 (2016)
46. Li, M., Cheng, W., Liu, K., He, Y., Li, X., Liao, X.: Sweep coverage with mobile sensors. *IEEE Trans. Mob. Comput.* **10**(11), 1534–1545 (2011)
47. Li, Z., Liu, Y., Li, M., Wang, J., Cao, Z.: Exploiting ubiquitous data collection for mobile users in wireless sensor networks. *IEEE Trans. Parallel Distrib. Syst.* **24**(2), 312–326 (2013)
48. Arumugam, M.U.: Infuse: a TDMA based reprogramming service for sensor networks. In: *Proceedings of ACM SenSys*, pp. 281–282, Nov 2004

49. Krasniewski, M.D., Panta, R.K., Bagchi, S., Yang, C.L., Chappell, W.J.: Energy-efficient on-demand reprogramming of large-scale sensor networks. *ACM Trans. Sens. Netw.* **4**(1), 2 (2008)
50. Naik, V., Arora, A., Sinha, P., Zhang, H.: Sprinkler: a reliable and energy efficient data dissemination service for wireless embedded devices. In: *Proceedings of IEEE RTSS*, Dec 2005
51. Levis, P., Culler, D.: The firecracker protocol. In: *Proceedings of ACM SIGOPS European workshop*, p. 3, Sept 2004
52. Huang, L., Setia, S.: Cord: energy-efficient reliable bulk data dissemination in sensor networks. In: *Proceedings of IEEE INFOCOM*, pp. 574–582, Apr 2008
53. Maia, G., Guidoni, D.L., Aquino, A.L.L., Loureiro, A.A.F.: Improving an over-the-air programming protocol for wireless sensor networks based on small world concepts. In: *Proceedings of ACM MSWiM*, pp. 261–267, Oct 2009
54. Panta, R.K., Khalil, I., Bagchi, S.: Stream: low overhead wireless reprogramming for sensor networks. In: *Proceedings of IEEE INFOCOM*, pp. 928–936, May 2007
55. Rossi, M., Bui, N., Zanca, G., Stabellini, L., Crepaldi, R., Zorzi, M.: Synapse++: code dissemination in wireless sensor networks using fountain codes. *IEEE Trans. Mob. Comput.* **9**(12), 1749–1765 (2010)
56. MacKay, D.J.C.: Fountain codes. *IET Commun.* **152**(6), 1062–1068 (2005)
57. Hagedorn, A., Starobinski, D., Trachtenberg, A.: Rateless deluge: over-the-air programming of wireless sensor networks using random linear codes. In: *Proceedings of ACM/IEEE IPSN*, pp. 457–466, Apr 2008
58. Wang, X., Wang, J., Xu, Y.: Data dissemination in wireless sensor networks with network coding. *EURASIP J. Wireless Commun. Netw.* **2010**(1), 465–915 (2010)
59. Firooz, M.H., Roy, S.: Data dissemination in wireless networks with network coding. *IEEE Commun. Lett.* **17**(5), 944–947 (2013)
60. dos Santos Ribeiro, N., Jr., Tavares, R.C., Vieira, M.A.M., Vieira, L.F.M., Gnawali, O.: Cod-eDrip: improving data dissemination for wireless sensor networks with network coding. *Ad Hoc Netw.* **54**, 42–52 (2017)
61. dos Santos Ribeiro, N., Jr., Tavares, R.C., Vieira, M.A.M., Vieira, L.F.M., Gnawali, O.: Cod-eDrip: data dissemination protocol with network coding for wireless sensor networks. In: *Proceedings of EWSN*, pp. 34–49 (2014)
62. Cao, Z., Wang, J., Liu, D., Zheng, X.: Chase: taming concurrent broadcast for flooding in asynchronous duty cycle networks. In: *Proceedings of IEEE ICNP* (2016)
63. Jin, M., He, Y., Zheng, X., Fang, D., Xu, D., Xing, T., Chen, X.: Smoggy-link: fingerprinting interference for predictable wireless concurrency. In: *Proceedings of IEEE ICNP* (2016)
64. Liu, D., Hou, M., Cao, Z., He, Y., Ji, X., Zheng, X.: COF: exploiting concurrency for low power opportunistic forwarding. In: *Proceedings of IEEE ICNP* (2015)
65. Cao, Z., Liu, D., Wang, J., Zheng, X.: Chase: taming concurrent broadcast for flooding in asynchronous duty cycle networks. *IEEE/ACM Trans. Netw.* **25**(5), 2872–2885 (2017)
66. Ji, X., He, Y., Wang, J., Wu, K., Liu, D., Yi, K., Liu, Y.: On improving wireless channel utilization: a collision tolerance-based approach. *IEEE Trans. Mob. Comput.* **16**(3), 787–800 (2017)
67. Michael, K., Roger, W.: The capture effect in fm receivers. In: *Proceedings of DCOSS*, pp. 206–215 (2016)
68. Leentvaar, K., Flint, J.: The capture effect in fm receivers. *IEEE Trans. Commun.* **24**(5), 531–539 (1976)
69. Whitehouse, K., Woo, A., Jiang, F., Polastre, J., Culler, D.: Exploiting the capture effect for collision detection and recovery. In: *IEEE EmNetS-II*, pp. 45–52, May 2005
70. Wang, Y., He, Y., Mao, X., Liu, Y., Li, X.: Exploiting constructive interference for scalable flooding in wireless networks. *IEEE/ACM Trans. Netw.* **21**(6), 1880–1889 (2013)
71. Wang, Y., Liu, Y., He, Y., Li, X., Cheng, D.: Disco: improving packet delivery via deliberate synchronized constructive interference. *Trans. Parallel Distrib. Syst.* **26**(3), 713–723 (2015)
72. Ferrari, F., Zimmerling, M., Thiele, L., Saukh, O.: Efficient network flooding and time synchronization with glossy. In: *Proceedings of IEEE IPSN*, pp. 73–84, Apr 2011



73. Maróti, M., Kusy, B., Simon, G., Lédeczi, Á.: The flooding time synchronization protocol. In: Proceedings of ACM SenSys, pp. 39–49, Nov 2004
74. Ephremides, A., Truong, T.V.: Scheduling broadcasts in multihop radio networks. *IEEE Trans. Commun.* **38**(4), 456–460 (1990)
75. Ferrari, F., Zimmerling, M., Mottola, L., Thiele, L.: Low-power wireless bus. In: Proceedings of ACM SenSys, pp. 1–14, Nov 2012
76. Doddavenkatappa, M., Chan, M.C. Leong, B.: Splash: fast data dissemination with constructive interference in wireless sensor networks. In: Proceedings of USENIX NSDI, pp. 269–282, Apr 2013
77. Du, W., Liando, J.C., Zhang, H., Li, M.: When pipelines meet fountain: fast data dissemination in wireless sensor networks. In: Proceedings of ACM SenSys, pp. 365–378, Nov 2015
78. Du, W., Liando, J.C., Zhang, H., Li, M.: Pando: fountain-enabled fast data dissemination with constructive interference. *IEEE/ACM Trans. Netw.* **PP**(99), pp. 1–14 (2016)
79. Sun, J.Z.: Dissemination protocols for reprogramming wireless sensor networks: a literature survey. In: Proceedings of IEEE SENSORCOMM, pp. 151–156, July 2010
80. Dong, W., Chen, C., Liu, X., Teng, G., Bu, J., Liu, Y.: Bulk data dissemination in wireless sensor networks: modeling and analysis. *Comput. Netw.* **56**(11), 2664–2676 (2012)
81. Li, Z., Li, M., Liu, J., Tang, S.: Understanding the flooding in low-duty-cycle wireless sensor networks. In: Proceedings of ICPP (2011)
82. Parthasarathy, R., Shirazi, B.A., Peterson, N., Song, W.Z., Hurson, A.: Management and security of remote sensor networks in hazardous environments using over the air programming. *Inf. Syst. e-Bus. Manag.* **10**(4), 521–548 (2012)
83. Tan, H., Ostry, D., Zic, J., Jha, S.: A confidential and dos-resistant multi-hop code dissemination protocol for wireless sensor networks. *Comput. Security* **32**, 36–55 (2013)
84. Wang, G., Zou, Y., Zhou, Z., Wu, K., Ni, L.M.: We can hear you with WiFi!. *IEEE Trans. Mob. Comput.* **15**(11), 2907–2920 (2016)
85. Zou, Y., Wang, G., Wu, K., Ni, L.M.: SmartScanner: know more in walls with your smartphone!. *IEEE Trans. Mob. Comput.* **15**(11), 2865–2877 (2016)
86. Peng, S., Li, S.S., Liao, X.K., Peng, X.Y., Xiao, N.: A scalable code dissemination protocol in heterogeneous wireless sensor networks. *Sci. China Inf. Sci.* **55**(6), 1323–1336 (2012)
87. Guo, X., Zheng, X., He, Y.: WiZig: cross-technology energy communication over a noisy channel. In: Proceedings of IEEE INFOCOM (2016)
88. Yin, S., Li, Q., Gnawali, O.: Interconnecting WIFI devices with IEEE 802.15.4 devices without using a gateway. In: Proceedings of DCOSS (2015)
89. Chebrolu, K., Dhekne, A.: Esense: communication through energy sensing. In: Proceedings of ACM MobiCom (2009)
90. Zheng, X., He, Y., Guo, X.: StripComm: interference-resilient cross-technology communication in coexisting environments. In: Proceedings of IEEE INFOCOM (2018)
91. Guo, X., He, Y., Zheng, X., Yu, L., Gnawali, O.: ZigFi: harnessing channel state information for cross-technology communication. In: Proceedings of IEEE INFOCOM (2018)
92. Kim, S.M., He, T.: Freebee: cross-technology communication via free side-channel. In: Proceedings of ACM MobiCom (2015)
93. Liu, Z., He, T.: WEBee: physical-layer cross-technology communication via emulation. In: Proceedings of ACM MobiCom (2017)