Improving Energy Adaptivity of Constructive Interference-Based Flooding for WSN-AF

Dapeng Cheng,¹,² Yanyan Mao,²,³ Yin Wang,⁴ and Xiangrong Wang¹

¹College of Information Science and Engineering, Shandong University of Science and Technology, Qingdao 266590, China
²School of Computer Science and Technology, Shandong Institute of Business and Technology, Yantai 264005, China
³School of Geosciences, China University of Petroleum, Qingdao 266580, China
⁴TNLIS T, School of Software, Tsinghua University, Beijing 100084, China

Correspondence should be addressed to Yin Wang; wangyin00@gmail.com

Received 30 October 2014; Revised 4 March 2015; Accepted 20 March 2015

Copyright © 2015 Dapeng Cheng et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Constructive interference (CI) is a synchronous transmission technique for multiple senders transmitting the same packet simultaneously in wireless sensor networks (WSNs). CI enables fast and reliable network flooding in order to reduce the scheduling overhead of MAC protocols, to achieve accurate time synchronization, to improve link quality of lossy links, and to realize efficient data collection. By achieving microsecond level time synchronization, Glossy realizes millisecond level CI-based flooding and 99% reliability. However, Glossy produces substantial unnecessary data forwarding, which significantly reduces the network lifetime. This is a very critical problem, especially in energy-limited large-scale wireless sensor networks for agriculture and forestry (WSN-AF) system. In this paper, we present an energy adaptive CI-based flooding protocol (EACIF) by exploiting CI in WSN-AF. EACIF proposes a distributed active nodes selection algorithm (ANSA) to reduce redundant transmissions, thereby significantly reducing energy consumption and flooding latency. We estimate the performance of EACIF both with real data traces and with uniformly distributed topology. Simulation results show that EACIF achieves almost the same packet reception ratio (PRR) as Glossy (e.g., 99%), while reducing 63.96% energy consumption. EACIF also reduces 25% flooding latency. When the packet interval is 30 seconds, EACIF achieves 0.11% duty cycle.

1. Introduction

With the emergence of the Internet of Things (IoT) [1, 2], wireless sensor networks for agriculture and forestry (WSN-AF) are becoming a research hotspot. WSN-AF systems can provide real-time field data, including environment temperature, relative humidity (RH), O₂ concentration, CO₂ concentration [3], and ethylene (C₂H₄) concentration. The implementation of WSN-AF system demands forwarding data [4], synchronizing sensor nodes [5], creating data collection tree [6], locating the monitored objects [7], reprogramming the revised code [8], and so forth. All these common applications in WSNs depend on the service of network flooding which propagates a packet through the whole network.

Power consumption, latency, and reliability are three critical factors of flooding in wireless sensor networks (WSNs). Recently, constructive interference- (CI-) based flooding is emerging as a latency optimal and high reliability flooding technique. As a representative CI-based flooding protocol, Glossy [9] realizes 99th percentile packet reception ratio (PRR). Real testbed (MoteLab, Twist, and Local) experiments show that Glossy achieves millisecond level flooding latency which is almost 100 times faster than traditional flooding solutions [10, 11]. Then a question arises: is Glossy energy-efficient? Unfortunately, Glossy makes all nodes involved in the data forwarding, which directly triggers enormous number of superfluous data transmissions, eventually leading to unnecessary energy consumption and network life reduction. This is a very critical problem, especially in energy-limited large-scale WSNs. One effective solution is to reduce redundant transmissions while maintaining the advantages of CI-based flooding.

A concurrent transmission scenario of CI-based flooding can be illustrated in Figure 1. In this example, S1, S2, S3, S4, S5...
and R1 are randomly deployed in a square area. Suppose all nodes having the same transmission range. According to the overlapping relations shown in Figure 1, the adjacency sets of S1, S2, S3, S4, S5 can be expressed as Ns1 = {S5, S4, R1}, Ns2 = {S3, S4, R1}, Ns3 = {S2, S5, R1}, Ns4 = {S2, S1, R1}, and Ns5 = {S1, S3, R1}. As the most common behavior of CI-based flooding, S1, S2, S3, S4, S5 are forwarding an identical packet to R1 simultaneously. In order to transfer as soon as possible, Glossy makes all the five senders in the radio on mode (dashed arrows in Figure 1) which is obviously unnecessary. In fact, an optimized forwarding set can be selected via lightweight communication between neighbors. Suppose that the residual energy of each node can be expressed as Er1, Er2, Er3, Er4, and Er5. The residual energy of each candidate sender and its neighbors is as follows: Es1 = {Er1, Er5, Er4}, Es2 = {Er2, Er3, Er4}, Es3 = {Er2, Er3, Er5}, Es4 = {Er1, Er5, Er4}, and Es5 = {Er1, Er3, Er5}. Then we have Er1 > Er2 > Er3 > Er4 > Er5. Each node independently selects the node which has the largest residual energy from Esi (i = 1, ..., 5). For the example in Figure 1, S1 and S2 are selected as the final senders. S3, S4, and S5 can keep silent to save energy. As shown by the red arrows in Figure 1, we can achieve the same flooding performance but save 60% energy consumption compared with Glossy. In particular, all nodes have the same residual energy in the initial state. There are two disjoint forwarding sets {S1, S2} and {S4, S5}. Network lifetime can be greatly extended by energy adaptive scheduling {S1, S2} (red arrows in Figure 1) or {S4, S5} (green arrows in Figure 1) as the practical forwarding set.

Inspired by this insight, we propose an energy adaptive CI-based flooding protocol (EACIF) for WSN-AF. EACIF improves energy adaptive CI-based flooding through energy scheduling and topology control. Compared with the state-of-art CI-based flooding protocols, EACIF achieves lower energy consumption, lower latency, and the same coverage and reliability. The major contributions of this work are summarized as follows.

(i) We are the first to propose a lightweight distributed energy adaptive CI-based flooding protocol (EACIF) for WSNs. EACIF adapts to the network changes in residual energy and selects the optimized forwarding set via the communication between neighbors.

(ii) EACIF constructs a sparse topology for CI-based flooding via an approximate minimum connected dominating set (MCDS) algorithm, which can be completed approximately in O(n).

(iii) EACIF proposes a k-covered mode switching algorithm (KMSA) for k-covered UDG and theoretically derives the energy saving formula for CI-based flooding.

(iv) We simulate EACIF based on uniformly distributed topology and real topology. Simulation results show that EACIF can achieve the same reliability as Glossy, while reducing 63.96% energy consumption on the real trace.

The rest of the paper is organized as follows. We review the related work in Section 2. In Section 3, we analyze the process of CI-based flooding and present active nodes selection algorithm (ANSA) and k-covered mode switching algorithm (KMSA). We evaluate the performance of EACIF in Section 4 and conclude the paper in Section 5.

2. Related Work

Constructive interference (CI) is a physical layer phenomenon and was first experimentally discovered by Dutta et al. [12]. Subsequently, CI is used in backcast [13] to avoid broadcast storm problem. CI can be achieved by sub-microsecond time synchronization of all coordinated senders. A typical scenario of CI is shown in Figure 2: S1, S2, and R all employ IEEE 802.15.4 radios. The packets (Ps1 and Ps2) with the same data are, respectively, transmitted from S1 and S2 to R. R can decode the Ps1 and Ps2 as one packet if the maximum temporal displacement Δ of Ps1 and Ps2 is no more than one chip period Tc = 0.5 μs. By implementing chip-level time synchronization on IEEE 802.15.4 radio, CI can greatly improve the efficiency of concurrent transmission in WSNs. Compared with the typical concurrent transmission technologies, capture effect (CE) [14] and message-in-message (MIM) [15], CI neither needs to set packet with strong signal
CI requires that multiple senders simultaneously transmit the same packet. This behavior is consistent with the characteristics of network flooding. Besides CI-based flooding, opportunistic flooding [17] also achieves fast network flooding by constructing a tree-based topology. Previous technologies such as CF [11] and RBP [10] schedule the data forwarding through link quality assessment. Moreover, when multiple sensor nodes simultaneously send an identical packet to the same destination, existing flooding solutions (RBP [10], FLASH [18], and CF [11]) need to dispatch the transmission order. Otherwise, it will cause mutual interference. The dispatch overhead should never be neglected in the large size of WSN-AF system. Fortunately, CI-based flooding can eliminate redundant dispatch overhead while increasing the flooding speed.

Glossy [9] achieves the synchronization condition ($\Delta \leq 0.5 \mu s$) of CI by capturing interrupts of IEEE 802.15.4 radio. Accurate synchronization guarantees high reliability of CI-based flooding. However, Glossy produces considerable redundant packets during the flooding process. This leads to massive unnecessary energy consumption. LWB [19], Chaos [20], and Choco [21] build up level scheduling mechanism for data collection or dissemination based on Glossy. The above works [19–21] achieve low duty cycle and efficient network flooding. However, they do not fundamentally change the transmission mechanism of Glossy which brings unnecessary energy consumption. Splash [22] forms a parallel pipeline by scheduling channel switch between nodes on adjacent layers. Splash can achieve higher throughput than Glossy. However, frequent channel switching increases the cumulative synchronization error which decreases synchronization accuracy and reliability. Meanwhile, channel scheduling brings more energy consumption.

Recently, Wang et al. [23] prove that Glossy has a scalability problem. The PRR of Glossy is inversely proportional to the hops of independent paths. The reason is that independent paths increase cumulative synchronization errors. For example, the GreenOrbs project [3] is designed to deploy 1000 nodes in Tianmu Mountain, Jiaxing, China, for forest carbon sink. In order to achieve accurate data acquisition, GreenOrbs deploys 8–20 neighbors (according to the diversity of node’s communication range) for every sensor node. The drawback of scalability problem is more serious in such a high dense WSN-AF system. Wang et al. also propose SCIF protocol which leverages grid topology to reduce the number of independent paths. However, the grid topology increases the path length of CI-based flooding and therefore increases the network delay. CX [24] uses the changes of relay count to conduct link selection, to some extent, reducing the number of nodes involved in transmission. Due to the randomness of relay count, CX costs considerable computational overhead and energy consumption. Compared with the above CI-based flooding protocols, EACIF is an energy adaptive CI-based flooding mechanism. EACIF achieves high reliability, high speed, and low-power network flooding via distributed management of node residual energy.

3. EACIF Implementation

Enlightened by SCIF [23], we find that a sparse topology can help to achieve energy adaptive CI-based flooding. However, there are still some challenges. The first challenge is to guarantee full network coverage while achieving fast network flooding and high packet reception ratio (PRR). The second challenge is to achieve a lightweight distributed energy scheduling algorithm for redundancy cover sets. In this section, we introduce the implementation of EACIF. We analyse the state migration of CI-based flooding in Section 3.1. Then, we introduce the topology control algorithm in Section 3.2, followed by the design of $k$-covered node switching algorithm in Section 3.3. Moreover, we theoretically analyze the energy consumption of EACIF in Section 3.4.

3.1. The Analysis of CI-Based Flooding

CI-based flooding is a network flooding technique based on the hierarchical model. In the first round, the initiator broadcasts a packet to its one-hop neighbors. Then, the one-hop neighbors will forward the received packets to the initiator and the two-hop neighbors in the second round. Every other round, nodes on each layer forward the flooding packet once. Each node stops forwarding packet if its transmission count reaches the transmission threshold. Key factors of CI-based flooding are listed in Table 1. $Tx$ represents the node that starts to send a packet and $Rx$ is the specific event of receiving packet. When the flooding begins, only the initiator can send packet. Let $SF_j$ denote the node that becomes an initiator and $SF_F$ let denote the node that becomes a receiver. $SF_j$ and $SF_F$ events are only triggered at the beginning of the network flooding process. $Stop$ represents the external event stopping the network flooding. $CJ_x$ is the transmission counter which records the number of transmissions of the current node. $N_x$ is the transmission threshold. $Rx_f$ indicates that the node fails to receive the packet. We use $c$ to denote the relay count which will be increased by one after each successful reception.

As Figure 3 shows, CI-based flooding has four correlative states including $Radio$ off state, $Standby$ state, $Send$ state, and $Receive$ state. When $SF_j$ event happens, the initiator...
transforms from \textit{Radio off} state into \textit{Send} state. Then the initiator triggers the \textit{Tx} event to send a flooding packet to its one-hop neighbors. When \textit{SF}_\textit{f} event happens, the nodes except the initiator enter the \textit{Standby} state to wait for the incoming packets. When \textit{Rx} event happens, the node switches to \textit{Receive} state and starts to decode the received packet. If the reception is successful, \textit{Rx} is performed. Meanwhile, the node shifts to \textit{Send} state for new transmitting round. In contrast, if \textit{Rx}_\textit{f} happens (\(\Delta \geq 0.5\mu s\)), the node converts into \textit{Standby} state for the next receiving round. After \textit{Tx} event is completed, the node has two candidate states: the \textit{Standby} state and the \textit{Radio off} state. If \(C_{tx}\) is less than \(N_{fx}\), the node will turn to \textit{Standby} state to wait for the new \textit{Rx} event. If \(C_{fx}\) reaches \(N_{fx}\), the node will turn to \textit{Radio off} state.

Table 2 shows the energy consumption of a typical IEEE 802.15.4 compliant radio CC2420 [25]. The power consumption of \textit{Send} state and \textit{Receive} state is at milliampere level, while \textit{Radio off} state and \textit{Standby} state only consume microampere-level power. Obviously, CI-based flooding should convert the nodes to \textit{Send} state and \textit{Receive} state as seldom as possible. As a topology independent CI-based flooding protocol, Glossy employs all nodes in WSNs looping in \textit{Standby} \(\rightarrow\) \textit{Receive} \(\rightarrow\) \textit{Send} until flooding ended. Each cycle has two “milliampere-level” states and one “microampere-level” state. As Figure 3 shows, each node in EACIF has two interconvertible modes: the active mode and the passive mode. In active mode, the node has the same state conversion schedule with Glossy, while the node in passive mode need not forward the received packet. If it has received the packet successfully, the node will turn to \textit{Radio off} state to save energy. If the reception has failed, the node will turn to \textit{Standby} state for the next \textit{Rx} event. Obviously, the more nodes in passive mode are, the more energy can be saved.

The scenario shown in Figure 4 illustrates the flooding process of EACIF when \(N_{fx} = 1\). Suppose an even-distributed WSN which has 25 nodes. Here, node I is the initiator; \(A_1, A_2, A_3,\) and \(A_4\) are working in active mode, while \(P_1, \ldots, P_{20}\) are working in passive mode. When the 1st transmission round begins, \(I\) broadcasts a packet to its one-hop neighbors \(A_1, A_2, A_3, A_4, P_1, P_2, P_3,\) and \(P_4\). Upon receiving the packet, \(P_1, P_2, P_3,\) and \(P_4\) enter the \textit{Radio off} state to save energy. Only \(A_1, A_2, A_3,\) and \(A_4\) participate in the 2nd transmission round. All the peripheral nodes (\(P_5, \ldots, P_{20}\)) are working in passive mode. It is not difficult to calculate that there are 80\% nodes that can work in passive mode to save energy. However, Glossy makes all the 25 nodes work in active mode.

3.2. Topology Control Algorithm. To save energy, one of the main challenges is to find the fewest nodes in active mode to cover the whole WSN. Then, we need to design an energy-driven policy for mode scheduling. In order to complete these challenges, EACIF proposes a distributed algorithm to select the active nodes in a given topology and develops a mode scheduling policy based on residual energy.

3.2.1. Geometric Definitions. We assume a WSN consisting of \(n\) sensor nodes deployed in a two-dimensional plane. The
Input: Given a node $v$ of a unit disk graph $G(V,E)$, $v \in V$.
Output: The node type $\psi_v$, where $\Psi$ denotes the set of node type and $\psi_v \in \Psi$.
1. Suppose $QA(v)$, $TA(v)$, $RA(v)$, $RE_v$ respectively represent AD request, AD announcement, the AD response and the residual energy of $v$.
2. Let $N_v = \{N_1, N_2, \ldots, N_p\}$ be neighbors set of node $v$, where $p$ is $v$'s neighbors number;
3. Let $NE_v = \{NE_1, NE_2, \ldots, NE_m\}$ be the residual energy set of $N_v$.
4. Let $\sigma$ be the threshold of residual energy;
5. \text{//Set all the node type into PD.}
6. $\psi_v = 0$;
7. $c = 1$;
8. \text{//Node v initially apply to become a AD.}
9. if $QA(v) == True$ && $RE_v \geq \sigma$ then
10. \text{//Node v sends the residual energy to its one-hop neighbors.}
11. $RE_v \rightarrow N_v$;
12. end
13. \text{//Node v receives residual energy from its one-hop neighbors.}
14. else if $RA(v) == True$ then
15. for $i = 1 : m$ do
16. if $RE_v < NE$ then
17. break;
18. end
19. $c++$;
20. end
21. if $c == m$ then
22. $\psi_v = 1$;
23. \text{//Declaring v as an AD node.}
24. $TA(v) = True$;
25. end
26. end
27. return $\psi_v$;

Algorithm 1: AD election algorithm.

node set is defined as $V$. A set of edges $E$ can be defined by the Euclidean distance of any two nodes. In this way, we can define a simple undirected graph $G(V,E)$. Suppose that the communication radius of all nodes is equal to one unit, we have the following definitions.

**Definition 1.** In a simple undirected graph $G = (V,E)$, for all $u$, $v \in V$, if $(u,v) \in E$ and $\exists |uv| \leq 1$, then $G$ is a unit disk graph $UDG(V,E)$.

**Definition 2.** In UDG($V,E$), a subset $D$ of $V$ is a dominating set (DS) if any node $v$ in $V$ either is in $D$ or is a neighbor of some node $d$ in $D$.

**Definition 3.** In UDG($V,E$), a set $C \in V$ is a connected dominating set (CDS) if the subgraph derived by $C$ is a connected graph. For all $v \in C$, if $C - \{c\}$ is not a CDS, then $C$ is a minimal connected dominating set (MCDS).

**Definition 4.** A UDG($V,E$) is a $k$-covered UDG if it contains $k$ groups of mutually disjoint MCDS.

We call the nodes in $D$ “active dominators” (AD). The gateway nodes converting DS to CDS are called “active gateways” (AG). The nodes outside CDS are called “passive dominatees” (PD).

3.2.2. Active Nodes Selection Algorithm. Enlightened by Section 3.1, we consider to construct an approximate MCDS as a sparse backbone. In a previous work, Wan et al. [26] have proposed an effective MCDS generation algorithm for wireless ad hoc networks. However, Wan's approach does not optimize the residual energy. In this paper, we propose a distributed active nodes selection algorithm (ANSA) which generates an approximate MCDS according to node’s residual energy.

Definitions 2 and 3 indicate that “active nodes” include the AD nodes and the AG nodes. Therefore, ANSA can be divided into two subalgorithms: AD election subalgorithm and AG election subalgorithm.

**AD Election Subalgorithm.** AD election subalgorithm constructs a CDS of a given topology. As Algorithm 1 shows, $\psi_v$ denotes the type of node $v$. The type of $v$ is AD if $\psi_v$ equals 1, while $\psi_v = 0$ indicates that the node is a PD. If $v$ is a AG, $\psi_v$ equals 2. In the initial state, the type values of all nodes are zero. Any node $v$ must broadcast its request to its one-hop neighbors ($QA(v) == True$) if it wants to be an AG. The request packets contain the residual energy of node $v$. Then, its neighbor nodes ($N_v$) send response packets to $v$. When receiving the responses ($RA(v) == True$), $v$ compares its residual energy ($RE_v$) with the residual energy set $NE_v$. Node $v$ is elected as AG if and only if $v$ has the maximum residual energy.
Input: Given node \( v \) of a unit disk graph \( G(V, E) \).
Output: The node type \( \psi_r \), where \( \Psi \) denotes the set of node type and \( \psi_r \in \Psi \).

(1) Suppose \( \alpha \) and \( \rho \) respectively represent the sets of AD node and PD node;
(2) Suppose \( QG(u, v, w) \), \( RG(u, v, w) \), \( TG(u, v, w) \), \( TA(v) \) respectively represent \( v \)'s AG request, AG response, AG announcement and AD announcement;
(3) Let \( y(v) = \{u1, v1, (u2, v), \ldots, (uk, v)\} \) be the (AD, PD) pairs set of \( v \), \( k \leq 5 \);
(4) Let \( \alpha(u, w) \) be the connected relation set of AD nodes \( u \) and \( w \), where \( l \) is the number of elements of \( \alpha(u, w) \);
(5) Let \( N_p = \{N_1, N_2, \ldots, N_p\} \) be neighbors set of node \( v \), where \( p \) is \( v \)'s neighbors number;
(6) Let \( RE_v \) be the residual energy of \( v \), and \( \sigma \) be the threshold value of residual energy;
(7) //PD processing.
(8) if \( v \in \rho \) then
(9) //Node \( v \) broadcast the (AD, PD) pairs.
(10) if \( TA(v) == \text{True} \) then
(11) \( y(v) \rightarrow N_v \);
(12) end
(13) if \( u, w \) are AD nodes connected by \( v \).
(14) \( QG(u, v, w) == \text{True} \); if \( RG(u, v, w) == \text{True} \) then
(15) \( TG(u, v, w) == \text{True} \);
(16) \( \psi_r = 2 \);
(17) end
(18) return \( \psi_r \);
(19) end
(20) //AD processing.
(21) if \( v \in \alpha \) then
(22) if \( v \) received \( QG(v, x, w) \) of \( x \) then
(23) //\( x \) is the candidate AG of \( v \) and \( w \).
(24) Traverse \( \alpha(v, w) \);
(25) if \( RE_x \) has the residual energy \&\& \( RE_x \geq \sigma \) then
(26) \( RG(v, x, w) == \text{True} \);
(27) end
(28) end
(29) end

Algorithm 2: AG election subalgorithm.

energy. Finally, \( v \) sets its node type and triggers \( TA(v) \) event to send the declaring packets. After AD election subalgorithm, each node in \( G(V, E) \) is either a AG or a PG.

AG Election Subalgorithm. As Algorithm 2 shows, all PD nodes first broadcast their (AD, PD) pairs. Alzoubi et al. [27] have proved that a PD of UDG(V, E) has at most five AD nodes. After gathering two-hop AD nodes information, a PD node \( v \) can broadcast AG request (\( QG(u, v, w) \)) applying to be AD between \( u \) and \( w \) (\( u \) and \( w \) are AD nodes). Any two AD nodes may have an interval of one hop or two hops. If the interval is one hop (\( u \rightarrow v \rightarrow w \)) and \( v \) has the most residual energy, any of the two AD nodes (\( u \) or \( w \)) can trigger \( TG(u, v, w) \) event. If the interval is two hops, \( QG(u, v, w) \) and \( QG(u, z, w) \) are, respectively, sent to \( u \) and \( w \). If \( v \) and \( z \) have the maximum residual energy, they are both elected as AG nodes. Otherwise, \( u \) and \( w \) cooperatively select the AG nodes for three-hop path (\( u \rightarrow v \rightarrow z \rightarrow w \)) by Algorithm 2.

Time Complexity. As a distributed algorithm, the time complexity of a AD election algorithm is \( O(m) \). For a PD node, the time complexity of a AG election algorithm is \( O(m) \). For a AD node, the time complexity of a AG election algorithm is \( O(m) + O(l) \). Therefore, time complexity of ANSA can be approximated as \( O(n) \).

3.3. \( k \)-Covered Mode Switching Algorithm. According to Definition 4, a \( k \)-covered UDG can generate \( k \) groups of MCDS. \( k \)-covered mode switching algorithm (KMSA) motivates us to design an energy adaption strategy for node's residual energy. As Algorithm 3 shows, every node \( v \) of a \( k \)-covered UDG \( G(V, E) \) holds a predefined residual energy threshold \( \sigma \). If the residual energy of any node (generally AD node or AG node) is lower than \( \sigma \), KMSA is triggered for a new MCDS building round. KMSA first empties the AD and AG sets and resets all the nodes’ type to PD. Then, KMSA triggers Algorithms 1 and 2 to find new AD and AG sets. If the new \( \alpha \) and \( \beta \) are not NULL, KMSA generates new MCDS which prolongs the lifetime of WSNs. However, if KMSA cannot find the new AD set or AG set, it will switch the type of all nodes to AD.

3.4. Power Consumption Model. In this section, we first establish the model of power consumption for CI-based flooding and then analyze the energy saving by EACIF in \( k \)-covered UDG.

Referring to Section 3.1, we define the main specifications of CI-based flooding. The unit time power consumption of Standby state, Send state, and Receive state can be, respectively, described as \( E_{sd} \), \( E_{sx} \), and \( E_{rx} \). The durations of Standby state, Send state, and Receive are \( T_{sd} \), \( T_{sx} \), and
Input: Given a \( k \)-covered UDG \( G(V, E) \).

Output: The set of node type \( \Psi, \psi_i \in \Psi \).

1. Let \( \Psi \) be the set of node type and \( \psi_i \in \Psi \);
2. Let \( \alpha \) and \( \beta \) be the sets of \( AD \) node and \( AG \) node;
3. Let \( T_{AD} \) and \( T_{AG} \) be the triggers of Algorithms 1 and 2;
4. Let \( RE_v \) be the residual energy of \( v \) and \( \sigma \) be the threshold value of residual energy;
5. \( \text{if } \exists (RE_v < \sigma) \text{ then} \)
   \( \alpha = NULL \);
6. \( \beta = NULL \);
7. \( \forall \psi_i = 0 \);
8. \( T_{AD} = True \);
9. \( T_{AG} = True \);
10. \( \text{if } \alpha \neq NULL \text{ then} \)
11. \( \text{end} \)
12. \( \text{else} \)
13. \( \forall \psi_i = 1 \);
14. \( \text{end} \)
15. \( \text{if } \beta == NULL \text{ then} \)
16. \( \forall \psi_i = 1 \);
17. \( \text{end} \)
18. \( \text{end} \)
19. \( \text{return } \Psi \);

Algorithm 3: \( k \)-covered mode switching algorithm.

\( T_{rx} \). Other parameters include the packet length \( P_l \), the data transmission rate \( b \), and node number \( n \).

Glossy completes flooding after all nodes run \( N_{tx} \) cycles of \( \text{Standby} \rightarrow \text{Receive} \rightarrow \text{Send} \). The power consumption of Glossy (\( E_{glossy} \)) can be expressed as

\[
E_{glossy} = nN_{tx}(E_{tx} \times T_{tx} + E_{tx} \times T_{rx} + E_{rx} \times T_{rx})
\]

\[
= nN_{tx}(E_{tx} \times T_{tx} + E_{tx} \times \frac{P_l}{b} + E_{rx} \times \frac{P_l}{b})
\]

(1)

The energy consumption of active nodes is the same as Glossy, while the passive nodes only need to run \( \text{Standby} \rightarrow \text{Receive} \) once. The power consumption of EACIF (\( E_{eacif} \)) can be calculated as

\[
E_{eacif} = E_{active} + E_{passive}
\]

where

\[
E_{active} = cN_{tx}(E_{tx} \times T_{tx} + \frac{P_l}{b} (E_{tx} + E_{rx}))
\]

(2)

\[
E_{passive} = (n - c) (E_{tx} \times T_{tx} + \frac{P_l}{b} E_{rx})
\]

Here, \( c \) is the number of active nodes. \( E_{active} \) and \( E_{passive} \) are the energy consumption of active nodes and passive nodes, respectively. The energy saving of EACIF (\( E_{saving} \)) can be calculated as

\[
E_{saving}(c) = E_{glossy} - E_{eacif}
\]

\[
= E_{glossy} - E_{active} - E_{passive}
\]

\[
= (n - c) \left( (N_{tx} - 1) (E_{tx} \times T_{tx}) + N_{tx} \frac{P_l}{b} E_{tx} \right)
\]

\[
+ (N_{tx} - 1) \frac{P_l}{b} E_{rx} \right).
\]

(3)

Equation (3) shows that every passive node reduces \((N_{tx} - 1)\) \( \text{Standby} \), \((N_{tx} - 1)\) \( \text{Receive} \), and \((N_{tx} - 1)\) \( \text{Send} \) states. However, active nodes cannot save energy. EACIF can reduce energy consumption because it reduces the number of active nodes via ANSA algorithm. Moreover, suppose \( \Omega = \{c_1, c_2, \ldots, c_k\} \) is the active nodes set for a \( k \)-covered UDG. Here, \( c_i \) denotes the number of active nodes in every cover set. EACIF’s energy saving for \( k \)-covered UDG \( KE_{saving} \) can be calculated as

\[
KE_{saving} = \sum_{i=1}^{k} E_{saving}(c_i).
\]

(4)

Equation (4) indicates that the energy saving is proportional to the redundant MCDS number \( k \). Therefore, EACIF enables low-power network flooding in WSNs.

4. Performance

In this section, we evaluate EACIF in both real trace and uniformly distributed topology. Section 4.1 evaluates ANSA algorithm of Section 3.2. Section 4.2 measures the packet reception rate (PRR) of EACIF. Sections 4.3 and 4.4, respectively, evaluate the latency and energy consumption.
4.1. Topology Control Evaluation. We first evaluate the topology control algorithm of Section 3.2 by MATLAB 7.11 platform. We use a real trace of previous work [28] which has 1052 nodes deployed in a ten-kilometer area. As Figure 5(a) shows, all nodes labeled as PD are represented by yellow. Figure 5(b) illustrates the node type after ANSA. The AD nodes are represented by red, while green denotes the AG nodes. Evaluation shows that ANSA selects 83 AD nodes, 275 AG nodes, and 694 PD nodes. The nodes in the active mode are only 34.03%. It is indicated that there are 65.97% nodes working in the passive mode to save energy.

4.2. The PRR Evaluation. Figure 6 shows the evaluation of PRR. We define CI-based flooding parameters as follows: the length of packet payload is 32 bytes, the variance of clock frequency drift $\sigma_\rho = 5$ ppm, the threshold of time displacement $\Delta = 0.5 \mu s$ (for the IEEE 802.15.4 radio), and the period during one-hop packet reception and retransmission $T_{\text{slot}} = 1.124$ ms. In the real trace of Figure 5, we change the transmission distance from 40 m to 100 m. In the uniformly distributed topology, we increase the node number varying from 500 to 2000 (referring to the scale of existing WSN-AF system). All the simulation results are averaged by 100 times. As Figure 6(a) shows, the PRR of EACIF and Glossy are more than 97.1% ($N_{tx} = 1$). When $N_{tx} = 3$, both EACIF and Glossy achieve a 99.9% stability. Simulation results of uniformly distributed topology also prove that EACIF can achieve high PRR (see Figure 6(b)). However, we also note that there are lower PRR values in individual samples, especially when $N_{tx} = 1$. When the transmission
distance is 65 m ($N_{tx} = 1$), one sample of 100 samples has 76.4th percentage of PPR (see Figure 6(a)). The reason is that EACIF reduces the amount of retransmission and also reduces opportunities to obtain the packets. The nodes in passive mode can only receive the packets from their parent nodes, while Glossy gives each node at least two chances of receiving packets, if the network degree is more than 2. Fortunately, the average PRR of EACIF is close to Glossy.

### 4.3. Flooding Latency Evaluation.

For each node, the flooding latency is defined as the duration from SF$_i$ event to the first successful $R_x$ event (see Table 1). Figure 7(a) shows the flooding latency of 1052 nodes in real trace ($N_{tx} = 3$). In EACIF, the latency of 263 nodes is less than 1 ms and 11.2 ms is needed for all nodes to receive packets, while Glossy spends 25% time more on flooding the whole network.

Figure 7(b) plots the flooding latency of a uniformly distributed network with 2000 nodes. Using EACIF, 125-node latency is less than 1 ms. In uniformly distributed topology, Glossy spends 27% time more than EACIF. Although Glossy is a fast CI-based flooding protocol, our evaluation results show that EACIF can flood much more quickly. This is
because EACIF reduces retransmissions by constructing a sparse network topology. The simulation results indicate that EACIF can effectively resolve the scalability problem in large-scale WSNs.

4.4. Energy Consumption Evaluation. We use the average radio on time to evaluate the energy consumption of EACIF. Figure 8(a) shows the average radio on time of real trace. When the transmission distance is increased from 40 m to 100 m, the average radio on time of Glossy changes from 22.4 ms to 19.7 ms, while the average radio on time in EACIF changes from 15.1 ms to 7.1 ms. When the transmission distance is 100 m, EACIF can save 63.96% more energy consumption than Glossy. Figure 8(a) indicates that EACIF’s effect of energy saving is proportional to network density. Simulation results verify the indication of (4).

In Figure 8(b), the node number increases from 500 to 2000. The average radio on time of Glossy has 31.2% increase. It means that Glossy suffers scalability problem. However, the average radio on time of EACIF has a slight decrease. When the node number is 2000, the average radio on time of EACIF is 11.2 ms. We assume that a WSN-AF system sends a packet every 30 seconds. The duty cycle of EACIF is 0.11%.

5. Conclusions and Future Work

In this paper, we introduce EACIF, the first work to improve energy adaptivity of CI-based flooding. We first implement ASNA to construct a sparse backbone on a given topology. Then, we propose KMSA algorithm to control the flooding process. We experimentally validate the performance of EACIF on real trace and the uniformly distributed topology. Simulation results indicate that EACIF can efficiently reduce redundancy retransmissions and significantly reduce energy consumption. Our future work includes the performance measurements of EACIF in real-world large-scale WSN-AF and the exploitation of EACIF in wireless time synchronization and remote reprogramming.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

Acknowledgments

This work is partially supported by the National Natural Science Foundation of China (NSFC) under Grants nos. 61272426, 61373079, 61325013, 61033005, 61272430, 61173173, 61472227, 61272244, 61373175, 61750533, 61272369, and 11271007; the National Science Foundation of Shandong Province under Grants nos. ZR2011FL008 and ZR2011FL004, Specialized Research Fund for the Doctoral Program of Higher Education under Grants nos. 2013020120016 and 2012371810010, and the Fundamental Research Funds for the Central Universities of China under Project no. 2012dgz02 (Xi’an Jiaotong University).

References

[1] J. Han, C. Qian, P. Yang et al., “Geneprint: generic and accurate physical-layer identification for UHF RFID tags,” IEEE/ACM Transactions on Networking, 2015.
[2] J. Han, C. Qian, D. Ma, X. Wang, and J. Zhao, “Twins: device-free object tracking using passive tags,” in Proceedings of the IEEE Conference on Computer Communications (INFOCOM ’14), pp. 469–476, Toronto, Canada, April 2014.
[3] L. Mo, Y. He, Y. Liu et al., “Canopy closure estimates with GreenOrbs: sustainable sensing in the forest,” in Proceedings of the 7th ACM Conference on Embedded Networked Sensor Systems (SenSys ’09), pp. 99–112, November 2009.
[4] Z. Cao, Y. He, and Y. Liu, “L2: lazy forwarding in low duty cycle wireless sensor networks,” in Proceedings of the 31st Annual IEEE International Conference on Computer Communications (IEEE INFOCOM’12), pp. 1323–1331, Orlando, Fla, USA, March 2012.
[5] M. Maroti, B. Kusy, G. Simon, and A. Ledeczi, “The flooding time synchronization protocol,” in Proceedings of the 2nd ACM International Conference on Embedded Networked Sensor Systems (SenSys ’04), pp. 39–49, November 2004.
[6] J. Wang, Y. Liu, M. Li, W. Dong, and Y. He, “Qos: towards comprehensive path quality measurement in wireless sensor networks,” in Proceedings of the IEEE INFOCOM, pp. 775–783, Shanghai, China, April 2011.
[7] Z. Yang, Y. Liu, and X.-Y. Li, “Beyond trilateration: on the localizability of wireless ad-hoc networks,” in Proceedings of the IEEE INFOCOM, pp. 2392–2400, Rio de Janeiro, Brazil, April 2009.
[8] G. Tolle and D. Culler, “Design of an application-cooperative management system for wireless sensor networks,” in Proceedings of the 2nd European Workshop on Wireless Sensor Networks (EWSN ’05), pp. 121–132, Istanbul, Turkey, February 2005.
[9] F. Ferrari, M. Zimmerling, L. Thiele, and O. Saukh, “Efficient network flooding and time synchronization with Glossy,” in Proceedings of the 10th ACM/IEEE International Conference on Information Processing in Sensor Networks (IPSN ’11), pp. 73–84, April 2011.
[10] F. Stann, J. Heidemann, R. Shroff, and M. Z. Murtaza, “RBP: robust broadcast propagation in wireless networks,” in Proceedings of the 4th International Conference on Embedded Networked Sensor Systems (SenSys ’06), pp. 85–98, Boulder, Colo, USA, October 2006.
[11] T. Zhu, Z. Zhong, T. He, and Z. Zhang, “Exploring link correlation for efficient flooding in wireless sensor networks,” in Proceedings of the 7th USENIX Conference on Networked Systems Design and Implementation (NSDI ’10), p. 4, San Jose, Calif, USA, April 2010.
[12] P. Dutta, R. Musulioiu-e, I. Stoica, and A. Terzis, “Wireless ACK collisions not considered harmful,” in Proceedings of the 7th Workshop on Hot Topics in Networks (HotNets-VII), October 2008.
[13] P. Dutta, S. Dawson-Haggerty, Y. Chen, C.-J. M. Liang, and A. Terzis, “Design and evaluation of a versatile and efficient receiver-initiated link layer for low-power wireless,” in Proceedings of the 8th ACM International Conference on Embedded Networked Sensor Systems (SenSys ’10), pp. 1–14, Zurich, Switzerland, November 2010.
[14] K. Leenstra and J. H. Flint, “The capture effect in fm receivers,” IEEE Transactions on Communications, vol. 24, no. 5, pp. 531–539, 1976.
[15] N. Santhapuri, J. Manweiler, S. Sen, R. R. Choudhury, S. Nelakuduti, and K. Munagala, “Message in message (MIM): a case for reordering transmissions in wireless networks,” in *Proceedings of the HotNets-VII*, October 2008.

[16] Y. Wang, Y. He, D. P. Cheng, Y. Liu, and X. Li, “Triggercast: enabling wireless constructive collisions,” in *Proceedings of the IEEE INFOCOM*, Turin, Italy, April 2013.

[17] S. Guo, Y. Gu, B. Jiang, and T. He, “Opportunistic flooding in low-duty-cycle wireless sensor networks with unreliable links,” in *Proceedings of the 15th Annual ACM International Conference on Mobile Computing and Networking (MobiCom ’09)*, pp. 133–144, Beijing, China, September 2009.

[18] J. Lu and K. Whitehouse, “Flash flooding: exploiting the capture effect for rapid flooding in wireless sensor networks,” in *Proceedings of the IEEE 28th Conference on Computer Communications (INFOCOM ’09)*, pp. 2491–2499, Rio de Janeiro, Brazil, April 2009.

[19] F. Ferrari, M. Zimmerling, L. Mottola, and L. Thiele, “Low-power wireless bus,” in *Proceedings of the 10th ACM Conference on Embedded Networked Sensor Systems (SenSys ’12)*, pp. 1–14, November 2012.

[20] O. Landsiedel, F. Ferrari, and M. Zimmerling, “Chaos: versatile and efficient all-to-all data sharing and in-network processing at scale,” in *Proceedings of the 11th ACM Conference on Embedded Networked Sensor Systems*, p. 1, ACM, November 2013.

[21] M. Suzuki, Y. Yamashita, and H. Morikawa, “Low-power, end-to-end reliable collection using glossy for wireless sensor networks,” in *Proceedings of the 77th Vehicular Technology Conference (VTC ’13)*, pp. 1–5, June 2013.

[22] M. Doddavenkatappa, M. C. Chan, and B. Leong, “Splash: fast data dissemination with constructive interference in wireless sensor networks,” in *Proceedings of the 10th USENIX Conference on Networked Systems Design and Implementation (NSDI ’13)*, pp. 269–282, Lombard, Ill, USA, April 2013.

[23] Y. Wang, Y. He, X. Mao, Y. Liu, Z. Huang, and X. Li, “Exploiting constructive interference for scalable flooding in wireless networks,” in *Proceedings of the IEEE Conference on Computer Communications (INFOCOM ’12)*, pp. 2104–2112, Orlando, Fla, USA, March 2012.

[24] D. Carlson, M. Chang, A. Terzis, Y. Chen, and O. Gnawali, “Forwarder selection in multi-transmitter networks,” in *Proceedings of the 9th IEEE International Conference on Distributed Computing in Sensor Systems (DCoSS ’13)*, pp. 1–10, May 2013.

[25] CC2420 Datasheet, Texas Instruments, http://www.ti.com.cn/cn/lit/ds/symlink/cc2420.pdf.

[26] P. J. Wan, K. M. Alzoubi, and O. Frieder, “Distributed construction of connected dominating set in wireless ad hoc networks,” in *Proceedings of the IEEE INFOCOM*, pp. 1597–1604, June 2002.

[27] K. Alzoubi, X.-Y. Li, Y. Wang, P.-J. Wan, and O. Frieder, “Geometric spanners for wireless ad hoc networks,” *IEEE Transactions on Parallel and Distributed Systems*, vol. 14, no. 4, pp. 408–421, 2003.

[28] X. Mao, X. Miao, Y. He, X.-Y. Li, and Y. Liu, “CitySee: urban CO₂ monitoring with sensors,” in *Proceedings of the IEEE Conference on Computer Communications (INFOCOM ’12)*, pp. 1611–1619, Orlando, Fla, USA, March 2012.
