Energy-Efficient Reliable Data Dissemination in Duty-Cycled Wireless Sensor Networks

Kai Han†‡, Liu Xiang†, Jun Luo†, Mingjun Xiao‡, Liusheng Huang‡
†School of Computer Engineering, Nanyang Technological University, Singapore, 639798
‡School of Computer Science and Technology, University of Science and Technology of China, China, 230026

hankai@gmail.com, {xi0001iu, junluo}@ntu.edu.sg, {xiaomj, lshuang}@ustc.edu.cn

ABSTRACT
Because data dissemination is crucial to Wireless Sensor Networks (WSNs), its energy-efficiency and reliability are of paramount importance. While achieving these two goals together is highly non-trivial, the situation is exacerbated if WSN nodes are duty-cycled (DC) and their transmission power is adjustable. In this paper, we study the problem of minimizing the expected total transmission power for reliable data dissemination (multicast/broadcast) in DC-WSNs. Due to the NP-hardness of the problem, we design efficient approximation algorithms with provable performance bounds for it.

Categories and Subject Descriptors
C.2.1 [Computer-Communication Networks]: Network Architecture and Design—Network topology

Keywords
Wireless sensor networks, data dissemination, multicast, broadcast, duty-cycle, energy-efficiency, reliability

1. INTRODUCTION
Designing effective data dissemination mechanisms for Wireless Sensor Networks (WSNs) is of paramount importance, as WSNs rely on data dissemination to carry critical commands or code updates from a sink to a set of (or all) nodes in the networks [7, 16]. Reliability and energy-efficiency are perhaps the most crucial requirements for data dissemination due to the notorious packet-losses in wireless communications and the limited power supply of sensor nodes.

Achieving reliability and energy-efficiency simultaneously is by no means trivial, and the problem gets even more challenging when modern WSNs’ features such as duty-cycling and power-adjustability are taken into account. On one hand, in a Duty-Cycled WSN (DC-WSN) [5], nodes usually switch between active/dormant states periodically, which imposes extra difficulty on attaining energy-efficiency even without regarding reliability [4, 12, 15]. On the other hand, nodes’ capability of adjusting the transmission power (tx-power) to several levels brings out a tradeoff between reliability and energy-efficiency, as a node can increase its tx-power to improve link reliability at the cost of higher energy consumption [9]. All these entangled features make it extremely challenging to design energy-efficient reliable data dissemination mechanisms for DC-WSNs.

Despite major efforts on energy-efficient data dissemination in wireless networks, a large body of them have taken the over-optimistic assumption of error-free wireless transmissions [1, 4, 8, 12, 14, 15, 17]. Other proposals considering unreliable links roughly follow two lines: one line bears the common objective of minimizing the expected tx-power consumption under guaranteed reliability, but they are designed only for Always-Active Wireless Networks (AAWN-s) [2, 9]. This batch of work has unanimously concentrated on unicasting or employing unicasting to achieve multicasting [9–11] without leveraging the Wireless Broadcast Advantage (WBA). Another line aims at reducing the tx-power consumption for broadcasting in DC-WSNs without a stringent requirement on reliability. As a representative proposal in this line, [3] proposed a tree-based opportunistic flooding approach. However, the broadcast tree used in [3] is constructed for very low duty-cycle, hence per-hop transmissions are still done through unicasting. Pessimistically neglecting the WBA feature of wireless communication harms energy-efficiency. For example, suppose that nodes in Figure 1 all have a unit tx-power and A is the source node, then [9] creates the tree in Figure 1(b) for broadcast, whose expected tx-power is 19. Considering WBA leads to a better solution in Figure 1(c), whose expected tx-power is at most 12.9 (computed by Lemma 1 in Section 2).

In this paper, we present the first study on energy-efficient reliable data dissemination in DC-WSNs with guaranteed performance. We aim to minimize the expected total tx-power consumption in multicasting or broadcasting under guaranteed reliability, and we take into account WBA, unreliable links, power-adjustability and duty-cycling holistically. Given that the resulting Energy efficient Reliable Data Dissemination (ERDD) problem is NP-hard, we propose approximation algorithms for it with performance ratios of $O(\ln \Delta \ln k)$, where $\Delta$ is the maximum node degree in the network and $k$ is the total number of nodes involved in a multicast/broadcast session. Due to the page limits, all the proofs of our lemmas/theorems are omitted.
2. MODEL AND PROBLEM DEFINITION

We assume a set $V$ of DC-WSN nodes, switch between active/sleeping states periodically. The set of active time slots in a working period of any node $u \in V$ is denoted by $\mathcal{A}(u) \subseteq I = \{1, 2, ..., |I|\}$, where $I$ is the length of the working period. Following [3, 4, 12, 15], we assume that all nodes in $V$ are time synchronized, and a node can wake up its transceiver to transmit data at any time slot, but can only receive data when it is active. We also assume that each node $u \in V$ can adjust its tx-power to several levels, which are $\xi_{\min} = \xi_1 \leq \xi_2 \leq ... \leq \xi_d = \xi_{\max}$. Let $\mathcal{L} = \{\xi_1, \xi_2, ..., \xi_d\}$. When $u$’s tx-power is adjusted to $l \in \mathcal{L}$, let the link quality $p_{uv}(l) \in (0, 1]$ denote the success ratio of data transmission on link $(u, v)$. Following [9], we assume that $p_{uv}(l)$ increases with $l \in \mathcal{L}$ and the link qualities considered in the network have a positive lower bound (denoted by $\Lambda$). Let $\mathcal{N}_u(l)$ denote the node set $\{v | v \in \mathcal{V} \wedge \xi(l) \wedge p_{uv}(l) \geq \Lambda\}$.

In a data dissemination session, there exists a source node $s \in V$ that needs to send data to a set of destination nodes $\mathcal{R} \subseteq \mathcal{V} \setminus \{s\}$. Let $k = |\mathcal{R}| + 1$, hence the session is a broadcast if $k = |\mathcal{V}|$; otherwise a multicast. A valid power assignment for such a session is a function $L: \mathcal{V} \rightarrow \mathcal{L}$, such that when each node $u \in \mathcal{V}$ is adjusted to the tx-power level $L(u)$, there exists a data dissemination tree $T$ spanning the nodes in $\mathcal{R} \cup \{s\}$; and the set of parent/children nodes of any node $v \in T$ must be contained in $\mathcal{N}_u(L(v))$ for $v$ to send data and to receive control messages. To conduct a data dissemination session in a DC-WSN, we need not only to find a valid power assignment $L$ and a data dissemination tree $T$, but also to select the transmission time slots of the forwarding nodes in $T$, in order to avoid transmitting data to sleeping nodes. To this end, we provide the definition of Viable Data-dissemination Solution (VDS) in Definition 1, where we denote the set of non-leaf nodes in $T$ by $\lambda(T)$, the set of children nodes of any node $u$ in $T$ by $\mathcal{C}_T(u)$, and the set $\{t | v \in \mathcal{N}_u(l) \wedge t \in \mathcal{A}(v)\}$ by $\partial_t(u,l)$:

**Definition 1** (VDS). Suppose that $T$ is a data dissemination tree under a valid power assignment $L$, and $S$ is a function that satisfies:

1. For any $t \in I$ and any $u \in \lambda(T)$, $S(u, t) \subseteq \partial_t(u, L(u)) \cap \mathcal{C}_T(u)$,
2. For any $u \in \lambda(T)$, $\mathcal{C}_T(u) \subseteq \bigcup_{t \in I} S(u, t),$

then $S$ is called a Viable Transmission Schedule for $T$ and $(L, T, S)$ is called a Viable Data-dissemination Solution (VDS).

Basically, VDS requires a node $u \in T$ to be responsible for sending data to nodes in the set $S(u, t)$ at time slot $t$. Due to the link unreliability, $u$ may have to retransmit several times (in different working periods) for all the nodes in $S(u, t)$ to receive the data (if $S(u, t) \neq \emptyset$). To understand how many retransmissions are needed for a forward node, we introduce Lemma 1:

**Lemma 1.** For any node $u \in V$, any power level $l \in \mathcal{L}$ and any set $\mathcal{Q} \subseteq \mathcal{N}_u(l)$, let $\chi_u(l, \mathcal{Q})$ be the random variable that denotes the number of transmissions by $u$ for all the nodes in $\mathcal{Q}$ to receive a data packet when $u$’s tx-power is $l$. If $\mathcal{Q} \neq \emptyset$, then we have:

$$E[\chi_u(l, \mathcal{Q})] = \sum_{i=0}^{\infty} \left[1 - \prod_{v \in \mathcal{Q}} (1 - (1 - p_{uv}(l))^i)\right]$$

Our objective is to find a VDS such that the expected total transmission energy of the forward nodes is minimized. We introduce the formal definition of this problem by Definition 2 and Definition 3:

**Definition 2** (Energy-Consumption Function). The energy-consumption function of a VDS $(L, T, S)$ is

$$\Psi(L, T, S) = \sum_{u \in \mathcal{V}} \sum_{t \in I} L(u) \cdot \chi_u(L(u), S(u, t))$$

**Definition 3** (ERDD Problem). Given a set $V$ of DC-WSN nodes, a source node $s \in V$, and a set of receiver nodes $\mathcal{R} \subseteq \mathcal{V} \setminus \{s\}$, the Energy-efficient Reliable Data Dissemination (ERDD) problem is to find a VDS $(L^*, T^*, S^*)$ such that $\Psi(L^*, T^*, S^*)$ is minimized.

Clearly, if we assume that all links are reliable and nodes are active at all time slots, then the ERDD problem degenerates to the min-energy broadcast/multicast problem in traditional AAWNs with perfect links, which is known to be NP-hard [1, 17]. Hence we have:

**Theorem 1.** The ERDD problem is NP-hard.

In the following section, we will propose approximation algorithms for the ERDD problem.

3. APPROXIMATION ALGORITHMS

A basic idea of our approximation algorithm design is to build a special data structure called the Time-Reliability-Power (TRP) Space, where data dimensions on time, reliability and power levels are all involved to facilitate our algorithm design and analysis. In this section, we will first introduce the concepts about the TRP space, and then present our algorithms in details.

3.1 Time-Reliability-Power Space

To build a TRP space, we first define a positive number $\gamma_{uv}(l)$ for any $u \in V$, $l \in \mathcal{L}$ and $v \in \mathcal{N}_u(l)$ as follows:

$$\gamma_{uv}(l) = \begin{cases} 1 - 1/\ln(1 - p_{uv}(l)), & p_{uv}(l) \in (0, 1) \\ 1, & p_{uv}(l) = 1 \end{cases}$$

Note that $\gamma_{uv}(l)$ is no less than 1 and decreases when $p_{uv}(l)$ increases, hence a larger $\gamma_{uv}(l)$ indicates a poorer link. Besides, since link qualities have a constant lower bound in practice [9], $\gamma_{uv}(l)$ has a constant upper bound. Based on this definition, we introduce the concept of TRP space and a weight assignment method in Definition 4 and Definition 5, respectively.
Definition 4 (TRP SPACE). For any \( u \in \mathcal{V} \), define the TRP Set of \( u \) as \( \vartheta(u) = \{ (u, t, r, l) | t \in \mathcal{T} \land R \subseteq \mathcal{T} \land (t, r, l) \in \bigcup_{v \in \mathcal{V}} (\gamma_{uv}(l)) \} \). Define \( \mathcal{U} = \bigcup_{u \in \mathcal{V}} \vartheta(u) \cup \mathcal{V} \). An access-relationship \( \mathcal{W} \) is any set of ordered 2-tuples wherein each 2-tuple consists of two elements from \( \mathcal{U} \). The tuple \( (\mathcal{U}, \mathcal{W}) \) is called a TRP space.

Definition 5 (Weight Assignment). Any \( x \in \mathcal{U} \) has a weight \( \varpi(x) \). If \( x \in \mathcal{V} \) then \( \varpi(x) = 0 \). If \( x = (u, t, r, l) \in (\mathcal{U} \setminus \mathcal{V}) \), then \( \varpi(x) = r \cdot l \). The weight of any \( y \subseteq \mathcal{U} \) is also denoted by \( \varpi(y) = \sum_{x \in y} \varpi(x) \).

Intuitively, the elements in \( \mathcal{U} \) can be viewed as weighted 4-dimensional (vector) nodes whose adjacent relationships are determined by an access-relationship \( \mathcal{W} \), while \( \mathcal{W} \) is defined by a given problem, as we shall do in Section 3.2 and 3.3.

Based on above definitions, we introduce the concept of "TRP path" in Definition 6.

Definition 6 (TRP Path). Given a TRP space \( (\mathcal{U}, \mathcal{W}) \), a sequence \( h = (x_1, x_2, ..., x_m) \) is called a TRP path from \( x_1 \) to \( x_m \) iff \( (x_i, x_{i+1}) \in \mathcal{W} \forall 1 \leq i < m \). The length of \( h \) is defined as
\[
\ell(h) = \left\{ \begin{array}{ll}
\sum_{i=2}^{m-1} \varpi(x_i) & m > 2 \\
0 & \text{otherwise}
\end{array} \right.
\]
Based on the length function \( \ell \), the shortest TRP path from \( x \in \mathcal{U} \) to \( y \in \mathcal{U} \) is the one with the minimum length, which is denoted by \( x \rightsquigarrow y \).

In our algorithm design, we will map the components in a TRP space \( (\mathcal{U}, \mathcal{W}) \) to a data-dissemination solution. Hence, we introduce some definitions for this mapping. The node-image of any \( x \in \mathcal{U} \) is defined as the node \( u \in \mathcal{V} \) such that \( x \in \vartheta(u) \cap \{ u \} \). The edge-image of any tuple \( (u, y) \in \mathcal{W} \) is defined as the directed edge \( (u, v) : u \in \mathcal{V}, v \in \mathcal{V} \) such that \( x \in \vartheta(u) \cup \{ u \} \) and \( y \in \vartheta(v) \cup \{ v \} \). Suppose that \( (\mathcal{U}_1, \mathcal{W}_1) \) is a sub-space of \( \mathcal{U}, \mathcal{W} \) (i.e., \( \mathcal{U}_1 \subseteq \mathcal{U}, \mathcal{W}_1 \subseteq \mathcal{W} \)), we define a mapping function \( \tilde{\vartheta} \) that \( \tilde{\vartheta}(\mathcal{U}_1, \mathcal{W}_1) \) is a directed graph constructed by all node-images of the elements in \( \mathcal{U}_1 \) and all edge-images of the tuples in \( \mathcal{W}_1 \), i.e., all nodes in \( \tilde{\vartheta}(\mathcal{U}_1, \mathcal{W}_1) \) are in \( \mathcal{V} \).

For the convenience of description, we introduce some other notations/definitions about the TRP space here. Given a TRP space \( (\mathcal{U}, \mathcal{W}) \) and any \( x, y \in \mathcal{U} \), we say \( x \) is accessible to \( y \) (or \( y \) is accessible from \( x \)) iff \( (x, y) \in \mathcal{W} \). Given any \( \mathcal{U}_1 \subseteq \mathcal{U} \), we define the element closure and relationship closure of \( \mathcal{U}_1 \) with respect to \( \mathcal{W} \) as \( \varrho_1(\mathcal{U}_1, \mathcal{W}) = \mathcal{U}_1 \cup \{ y | x \in \mathcal{U}_1 \land (x, y) \in \mathcal{W} \} \) and \( \varrho_2(\mathcal{U}_1, \mathcal{W}) = \{ (u, y) | u \in \mathcal{U}_1 \land (x, y) \in \mathcal{W} \} \). For any TRP path \( h = (x_1, ..., x_m) \) in \( (\mathcal{U}, \mathcal{W}) \), we define \( \text{in}(h) = \{ x_i | 1 \leq i \leq m - 1 \} \). If \( \mathcal{U}_1 \) contains all \( x_i : 1 \leq i \leq m \), we say \( \mathcal{U}_1 \) embraces \( h \).

In Algorithm 1, we first consider \( (\mathcal{U}, \mathcal{W}_M) \) as an undirected graph with node set \( \mathcal{U} \) and edge set \( \mathcal{W}_M \) and find an approximate Node Weighted Steiner Tree (NWST) with node set \( \mathcal{U}' \) (line 1), then we map the NWST to an approximate solution \( (L^*, S^* \subset L^*) \) for ERDD (lines 2-18). Roughly speaking, the idea for doing this is that we can find the tx-power and transmission schedule of any \( u \in \lambda(T^m) \) based on the NWST nodes in \( \vartheta(u) \) (lines 8-18). To make this idea work correctly, we need to add some elements to \( \mathcal{U}' \) (hence expand \( \mathcal{U}' \) to \( \tilde{\mathcal{U}} \) (lines 4-7), because there may exist \( u \in \lambda(T^m) \) such that \( \mathcal{U}' \cap \vartheta(u) = \emptyset \) according to rule M2. The correctness and performance ratio of Algorithm 1 will be proved in Section 4. The dominating running time of Algorithm 1 is spent in line 1, which is determined by the time complexity of the NWST algorithm [6]. So we get:

**Theorem 2.** The time complexity of the MC-ERDD algorithm is \( O(k^2 |\mathcal{V}|^2 \Delta^2) \).
3.3 Solving ERDD for the Broadcast Case

Although Algorithm 1 can also be used for broadcast, we introduce another algorithm with a better approximation ratio for the broadcast case in this section. Again, the first step is to construct a TRP space \( \langle U, W_B \rangle \), but the access-relationship \( W_B \) is defined differently from \( W_M \) according to the following rules:

B1: An element \((u_1, t_1, r_1, l_1) \in U \setminus V \) is accessible to another element \((u_2, t_2, r_2, l_2) \in U \setminus V \) iff the boolean expression \([u_1 \neq u_2] \wedge [u_2 \in \partial_t(u_1, l_1)] \wedge [r_1 \geq \gamma_{u_1, u_2}(l_1)] \wedge [r_1 \in N_0(u_2)] \) is true.

B2: An element \( x = (u_1, t_1, r_1, l_1) \in U \setminus V \) is accessible to \( u_2 \in V \) iff \( u_1 \neq u_2 \) and there exists \( y \in \partial(u_2) \) such that \((x, y) \in W_B \).

B3: For any \( u \in V \) and any \( y \in \partial(u) \), \((u, y) \in W_B \).

Note that \( W_B \) is not symmetric and hence we cannot run Algorithm 1 on \( \langle U, W_B \rangle \). Actually, B1-B3 are deliberately designed with desirable properties for our new approximation algorithm BC-ERDD (Algorithm 2), whose idea originates from the NWST algorithm in [6]. As we shall prove, by leveraging rules B1-B3, Algorithm 2 yields a better performance ratio than Algorithm 1 in the broadcast case.

Algorithm 2: BC-ERDD \((V, s, Z, U, W_B, L, A)\)

1. \( X \leftarrow \emptyset; \ Z \leftarrow V - \{s\} \)
2. while \(|Z| > 0\) do
   3. \( \text{Find } a \in U \text{ and a non-empty set } D \subseteq Z \text{ such that } (|D| \geq 2 \wedge a = s) = \text{true and } \operatorname{avg}(a, D) \text{ is minimized.} \)
   4. \( b \leftarrow \bigcup_{d \in D} \text{in}(a \rightarrow y) \)
   5. \( A \leftarrow A \cup X \)
   6. \( Z \leftarrow Z - g_1(A, W_B) \)
   7. if \( b \neq s \) then \( Z \leftarrow Z \cup \{b\} \)
   8. Let \( T^b \) be an arbitrary directed spanning tree of \((g_1(X, W_B), g_2(X, W_B))\) rooted at \( s \)
   9. \( \text{Set } L^b, S^b \text{ for the nodes in } T^b \) using the same method as lines 8-18 of Algorithm 1
10. return \((L^b, T^b, S^b)\)

In Algorithm 2, we use \( X \) to denote the subset of \( U \) which will be mapped to the forward nodes in broadcasting, and use \( Z \) to denote the set of nodes in \( V \) which are not accessible from any element in \( X \). At first, \( X \) is initialized to \( \emptyset \) and \( Z \) is initialized to \( V - \{s\} \). Then the algorithm uses a greedy strategy to expand \( X \) and reduce \( Z \) until \( Z = \emptyset \) (lines 2-7). In each iteration, we greedily select an element \( a \in U \) and a subset \( D \) of \( U \) such that

\[
\operatorname{avg}(a, D) = \left( \frac{\sigma(a) + \sum_{d \in D} \ell(a \rightarrow y)}{|D|} \right)
\]

is minimized (line 3), and then update \( X \) and \( Z \) accordingly (lines 4-7). Intuitively, \( \operatorname{avg}(a, D) \) denotes the average cost for “covering” the nodes in \( D \), which is analogous to the “cost effectiveness” measure in the greedy set-cover algorithm [13].

When \( X \) is finally determined, we use it to find an approximation solution \((L^b, T^b, S^b)\) in lines 8-9 based on a similar mapping process as that in Algorithm 1 (regarding \( X \) as \( \mathcal{R} \)). The time complexity of Algorithm 2 is given by:

Theorem 3. The time complexity of the BC-ERDD algorithm is \( \mathcal{O}(k^4 \Delta^2) \).

4. PERFORMANCE ANALYSIS

In this section, we prove the correctness and performance ratios of the algorithms proposed in Section 3. We shall first introduce an analysis method which is used for analyzing both the MC-ERDD and the BC-ERDD algorithms, and then give the detailed performance analysis of each algorithm.

4.1 A Method for Performance Analysis

From Lemma 1 and Definition 2 we can see that, for any VDS \((L, T, S)\), the expectation value \( \mathbb{E}[\Psi(L, T, S)] \) is a summation of infinite series, which is hard to calculate. This makes it hard to find the performance ratios of our algorithms. To bypass this problem, we introduce a surrogate function \( \phi \), which is defined as follows:

\[
\phi(L, T, S) = \sum_{u \in L} \sum_{t \in T} L(u) \cdot \max\{\gamma_{uv}(L(u)) | v \in S(u, t)\}
\]

It serves as an approximation to the expectation of \( \Psi \). Let \((L, T, S)\) denote the output of Algorithm 1 or Algorithm 2. We find the quantitative relationships between \( \phi \) and \( \Psi \) by Lemma 2 and Lemma 3:

Lemma 2. We have \( \mathbb{E}[\Psi(L, T, S)] \leq (\ln \Delta + 1) \cdot \phi(L, T, S) \) and \( \sigma[\Psi(L, T, S)] \leq \sqrt{2(\ln \Delta + 1) \phi(L, T, S)} \), where \( \sigma(\cdot) \) denotes the standard deviation.

Lemma 3. \( \phi(L^*, T^*, S^*) \leq (1 + \frac{1}{m \Delta}) \lambda \mathbb{E}[\Psi(L^*, T^*, S^*)] \)

Let \((L^*, T^*, S^*)\) be a VDS such that \( \phi(L^*, T^*, S^*) \) is minimized. Intuitively, \((L^*, T^*, S^*)\) is a solution in which each forwarding node selects the more reliable links for data transmission. Combining the above lemmas with the fact that \( \phi(L^*, T^*, S^*) \leq \phi(L^*, T^*, S^*) \), we immediately get:

Theorem 4. If \( \phi(L, T, S) \leq \beta \cdot \phi(L^*, T^*, S^*) \), then we have \( \mathbb{E}[\Psi(L, T, S)] \leq \alpha \beta \cdot \mathbb{E}[\Psi(L^*, T^*, S^*)] \) and \( \sigma[\Psi(L, T, S)] \leq \sqrt{\frac{\alpha \beta}{\lambda}} \mathbb{E}[\Psi(L^*, T^*, S^*)] \) where \( \alpha = (1 + 1/\ln 2) \lambda (\ln \Delta + 1) \).

Theorem 4 actually suggests a method for analyzing our algorithms, i.e., to find the performance ratios of Algorithm 1 and Algorithm 2, we only need to find their approximation ratios with respect to the surrogate function \( \phi \). In the following sections, we will analyze our algorithms based on this method.

4.2 Analyzing the MC-ERDD Algorithm

We first prove the correctness of Algorithm 1 in Lemma 4, and then prove in Lemma 5 that \( \phi(L^m, T^m, S^m) \) is within constant times of \( \varphi(U^\prime) \), which is the weight of the NWST we found in Algorithm 1. The proofs of these lemmas are based on the construction rules of \( \langle U, W_B \rangle \) as well as the mapping process employed in Algorithm 1 that maps an NWST to a VDS.

Lemma 4. \((L^m, T^m, S^m)\) is a VDS for multicast.

Lemma 5. \( \phi(L^m, T^m, S^m) \leq (2\varepsilon_{max}/\varepsilon_{min}) \cdot \varphi(U^\prime) \)

Next, we reveal the quantitative relationship between \( \varphi(U^\prime) \) and \( \phi(L^*, T^*, S^*) \) by Lemma 6. The main idea behind Lemma 6 is that, we can find a tree spanning \( R \cup \{s\} \)
in $\langle U, W_B \rangle$ whose weight is at most $(1 + \lambda)\phi(L^*, \hat{T}^*, \hat{S}^*)$, whereas the NWST algorithm we used in Algorithm 1 has an approximation ratio of 2 in $k$.

Lemma 6. $\varpi(U') \leq 2(1 + \lambda)\ln k \cdot \phi(L^*, \hat{T}^*, \hat{S}^*)$

Recall that $\xi_{\text{max}}, \xi_{\text{min}}$ are both pre-defined constants and $\lambda$ has a constant upper bound. Hence, combining Lemma 5, Lemma 6, and Theorem 4 yields:

Theorem 5. The expectation value and standard deviation of $\Psi(L^b, T^b, S^b)$ are within $9.8\eta$ and $13.9\eta$ times of $\mathbb{E}[\Psi(L^*, T^*, S^*)]$, respectively, where $\eta = \max_{x_{\text{max}}} \lambda/(1 + \lambda)(\ln \Delta + 1) \ln k = O(\ln \Delta \ln k)$.

4.3 Analyzing the BC-ERDD Algorithm

Based on similar reasoning to the proof of Lemma 4, we can also prove the correctness of Algorithm 2 based on the definition of $W_B$, as shown by Lemma 7:

Lemma 7. $\langle L^b, T^b, S^b \rangle$ is a VDS for broadcast.

Suppose that the while loop in lines 2-7 of Algorithm 2 executes for $q$ times. Let $Z_j = Z$ and $X_j = X$ after the $j$th while loop is executed. Let $D_j$ be the set $D$ found in the $j$th while loop. Clearly we have $\varpi(X_0) = 0$ and $|Z_0| = 0$. Indeed, the value $(\varpi(X_{j+1}) - \varpi(X_j))/|D_{j+1}|$ represents the “cost effectiveness” of the elements added to $X$ in the $(j+1)$th loop, which is bounded by Lemma 8:

Lemma 8. $(\varpi(X_{j+1}) - \varpi(X_j)) \cdot |Z_j| \leq |D_{j+1}| \cdot \phi(L^*, \hat{T}^*, \hat{S}^*) (0 \leq j \leq q - 1)$

Note that the elements in $X_j$ are finally mapped to the transmission schedules of the non-leaf nodes in $T^b$. Hence, based on Lemma 8 and the mapping process in Algorithm 2, we can get:

Lemma 9. $\phi(L^b, T^b, S^b) \leq (\xi_{\text{max}}/\xi_{\text{min}}) \cdot (2\ln k + 1) \cdot \phi(L^*, \hat{T}^*, \hat{S}^*)$

Combining Lemma 9 with Theorem 4, we get:

Theorem 6. The expectation value and standard deviation of $\Psi(L^b, T^b, S^b)$ are within $2.5\mu$ and $3.5\mu$ times of $\mathbb{E}[\Psi(L^*, T^*, S^*)]$, respectively, where $\mu = \max_{x_{\text{max}}} \lambda/(1 + \lambda)(2\ln k + 1) = O(\ln \Delta \ln k)$.

5. CONCLUSION

We have studied the energy-efficient reliable data dissemination problem in DC-WSNs with unreliable links. We seek to minimize the total expected tx-power consumption for reliable multicasting/broadcasting. Due to the NP-hardness of the problem, we have proposed approximation algorithms with provable performance ratios. To the best of our knowledge, these algorithms are the first, one on hand, to holistically take into account various aspects including duty-cycling, wireless broadcast advantage, unreliable links and power-adjustability, and on the other hand, to provide guaranteed performance bounds for energy-efficient reliable data dissemination in DC-WSNs.

Acknowledgments

This work was supported in part by the ArRF Tier 2 Grant ARC15/11, the Start-up Grant of NTU, the National Natural Science Foundation of China (NSFC) No.61103007, the National Grand Fundamental Research 973 Program of China No.2011CB302905, and the National Science and Technology Major Project No. 2011ZX03005-004-04.

6. REFERENCES