Low-Latency SINR-based Data Gathering in Wireless Sensor Networks

Dawei Gong and Yuanyuan Yang

Department of Electrical and Computer Engineering, Stony Brook University, Stony Brook, NY 11794, USA

***Abstract*—Data gathering is a fundamental operation for various applications of wireless sensor networks (WSNs), where sensor nodes sense information and forward data to a sink node via multi-hop wireless communications. Typically, data in a WSN is relayed over a tree topology to the sink for effective data gathering. A number of tree-based data gathering schemes have been proposed in the literature, most of which aim at maximizing network lifetime. However, the timeliness and reliability of gathered data are also of great importance to many applications in WSNs. To achieve low-latency, high-reliability data gathering in WSNs, in this paper, we construct a data gathering tree based on a reliability model, schedule data transmissions for the links on the tree and assign transmitting power to each link accordingly. Since the reliability of a link is highly related to its signal to interference plus noise ratio (SINR), the SINR of all the currently used links on the data gathering tree should be greater than a threshold to guarantee high reliability. We formulate the joint problem of tree construction, link scheduling and power assignment for data gathering into an optimization problem, with the objective of minimizing data gathering latency. We show the problem is NP-hard and divide the problem into two subproblems: Construction of a low-latency data gathering tree; Jointly link scheduling and power assignment for the data gathering tree. We then propose a polynomial heuristic algorithm for each subproblem and conduct extensive simulations to verify the effectiveness of the proposed algorithms. Our simulation results show that the proposed algorithms achieve much lower data gathering latency than existing data gathering strategies while guaranteeing high reliability.**

***Index Terms*—Wireless sensor networks (WSNs), data gathering, link scheduling, power assignment, SINR constraint.**

I. Introduction

Recently, wireless sensor networks (WSNs) have exhibited great potentials as a new information-gathering approach for many applications, such as structure monitoring, security surveillance, and wildlife preservation. Besides sensing interested information, the paramount task in WSNs is how to gather data from scattered sensors. Typical approaches for data gathering are to forward sensed data to a static data sink via a few selected relaying nodes or dynamic routing [1]. In more complex approaches, sensed data are aggregated or compressed at relaying nodes by exploring spatiotemporal correlation [2]–[4], which introduces extra delay and may not be applicable to all applications. Another approach is to employ a mobile collector that roams around the sensing field by moving sufficiently close to sensors so as to collect data from them via short-range or direct communications [5]. Though this approach can significantly reduce the energy consumption of multi-hop relaying, the limit on the velocity of mobile collectors makes it difficult to complete data gathering timely in large-scale WSNs.

Research supported by NSF grant number ECCS-0801438 and ARO grant number W911NF-09-1-0154.

On the other hand, constructing a tree rooted at the sink is a simple, yet effective approach for data gathering in WSNs, as no routing decisions need to be made at sensor nodes. In such a scheme, each node transmits its own data and the data received from its children to its parent. Several schemes have been proposed for constructing trees for data gathering with the objective of prolonging network lifetime [1]–[3]. However, these schemes take an unrealistic assumption that all transmissions over the data gathering tree are always successful. In fact, it has been shown via experiments that a significant percent of links in WSNs are unreliable and asymmetric even under interference-free scenarios [6]. In other words, packets may need to be retransmitted multiple times over such links before they can be successfully delivered, which leads to extra latency and energy consumption in data gathering.

In addition, the medium access mechanism is highly related to the capacity and reliability of wireless links on the data gathering tree. However, it was seldom considered in aforementioned tree-based data gathering schemes. Carrier sense multiple access with collision avoidance (CSMA/CA) is a medium access control (MAC) mechanism commonly used in WSNs, but it cannot ensure high reliability and low latency. The reason is that with CSMA/CA, each node transmits packets in an opportunistic manner, such that severe interference or collisions could occur when two nearby links choose the same backoff interval and transmit simultaneously. Even though the reliability can be achieved by retransmissions at the MAC layer, this will lead to extra overhead and thus long latency for data gathering. In fact, the reliability of a link is closely related to its signal to interference-plus-noise ratio (SINR), which is defined as the ratio of received power from the transmitter of the link to the received power of all other links plus background noise. To guarantee high reliability, the SINR of a link should be greater than a threshold for the applied coding and modulation scheme. Time-division-multiple-access (TDMA) is another MAC mechanism for WSNs, where the MAC time is divided into time slots and links are active for transmission only in their assigned time slots, such that the SINR of each link in every time slot is deterministic. Given a set of wireless links, the problem of finding the minimum number of time slots such that traffic demands of all links are satisfied while the SINR of each link is greater than a threshold is called *link scheduling* problem, which is NP-hard [13]. This problem is further complicated when sensor nodes can transmit data at various power levels.

In this paper, we consider tree-based data gathering in WSNs.

978-1-4673-5946-7/13/$31.00 ©2013 IEEE

To guarantee high reliability, TDMA will be used by all sensor nodes to access the wireless medium. Specifically, we select a subset of links from the network to form a data gathering tree, schedule links on the tree to be active for data transmission in different time slots, and assign transmitting power to active links in each time slot, such that sensed data from all nodes are delivered to the sink reliably in the smallest number of time slots. We first formulate the problem into an optimization problem and prove its NP-hardness. Then to give practical solutions, we divide the problem into two subproblems: (1) Construct a low-latency data gathering tree; (2) Given a data gathering tree, find a link scheduling and transmitting power assignment strategy to collect sensed data fast and reliably. For the first subproblem, we propose a tree construction algorithm where each node selects its parent according to the load of relaying traffic and the introduced interference to other links that are already on the tree, among all potential candidates. For the second subproblem, we propose a joint link scheduling and transmitting power assignment algorithm which gives high priority to links that have heavy relaying traffic load or experience severe interference. As will be seen from our extensive simulation results, the proposed algorithms require much shorter time for data gathering than the compared schemes. The network throughput is further improved as packet retransmissions are reduced by guaranteeing high reliability of each transmission, making the proposed algorithms desirable candidates for data gathering in WSNs.

The remainder of the paper is organized as follows. Section II reviews the related work. Section III introduces the system model, provides the formulation of the data gathering problem and proves the NP-hardness of the optimization problem. Section IV divides the problem into two subproblems and presents two heuristic algorithms to solve the two subproblems. Section V presents the performance evaluation results for the proposed algorithms and Section VI concludes the paper.

II. Related Work

There has been extensive work in the literature on data gathering schemes in WSNs. Most of the work studied static data gathering and focused on maximizing network lifetime by taking advantage of data aggregation. A distributed protocol for constructing a data gathering tree was presented in [1], where each node maintains its own data gathering links based on local information. In [2], sensors are organized into clusters based on the correlation of their sensed data, to maximize the data aggregation level and minimize the number of packets to be sent to the sink. In addition, some work assumed that an intermediate node is capable of aggregating all received packets and its own sensed data into a single packet. Based on such an assumption, the problem of finding a maximum-lifetime data gathering tree in WSNs was proved to be NP-hard in [3] and an approximation algorithm was provided. In [4], an optimal algorithm was presented to find a maximum-lifetime tree from all shortest path trees rooted at the sink for data gathering by showing that it is equivalent to addressing a semi-matching problem between nodes from any two adjacent levels. However, the assumption of perfect data aggregation in these schemes may not hold in general WSNs. More importantly, the reliability of links on the tree was not considered in these schemes.

As discussed earlier, a TDMA-based MAC mechanism can be used to ensure link reliability, which divides the MAC time into time slots and schedules a subset of links to transmit in each time slot. There has been some work on link scheduling for data gathering in WSNs. The problem of finding a minimum length schedule for data gathering without aggregation was proved to be NP-hard in [8], and two distributed algorithms were proposed in [9] and [10], respectively. However, the interference model used in the above studies is inaccurate, as it determines a binary interfering relation for two links solely based on whether they are within a fixed interference range. In reality, the interference among multiple links may be too severe for all links to transmit data simultaneously even though any two of them are interference-free based on the interference model. This is because that interference from multiple transmitters is accumulated at each receiver. Consequently, links scheduled by these schemes may have poor reliability. Finally, a lower bound for minimum data gathering time was given in [11], under the assumption that assigning various transmitting power and channel to different nodes can eliminate the interference among links, which is not realistic. In fact, measurements in [12] have demonstrated that the SINR at receivers can characterize the interference among multiple links more precisely.

In the meanwhile, link scheduling under SINR interference model for general wireless networks has received much attention in recent years. It was proved [13] that the problem of minimum time link scheduling where all links satisfy a SINR threshold is NP-hard, even when all nodes are scattered in a Euclidean plane. Several approximation algorithms have been proposed in the literature [14]–[18]. However, these algorithms are based on an assumption that the traffic load on each link is independent, which does not hold for data gathering in WSNs, where many links help relaying traffic loads from other nodes to the sink node.

Furthermore, link scheduling under SINR interference model has been jointly studied with power control in the literature [19]–[22]. In [19], distributed power control under SINR interference model in WSNs was studied from the angle of game theory, showing that a Nash equilibrium can be reached by formulating the problem into a non-cooperative game. A power based interference graph was introduced in [20], in which if no feasible power assignment strategy can be found to ensure the SINR of any two links above a given threshold, the two corresponding vertices in the graph are connected. Although the interference graph cannot tell whether multiple links can be scheduled in the same time slot, it can sufficiently indicate two links cannot be scheduled concurrently if they are connected in the graph. In [22], Perron-Fronbenius eigenvalue condition was used as a bridge to tie link scheduling and power control together in an integrated manner. In the scheme, given a set of nonadjacent links, the feasibility of concurrent scheduling and the related power control vector can be obtained together by examining the eigenvalues and eigenvector of a matrix composed of channel gain ratios among these links. In this paper, we will adopt Perron-Fronbenius condition as a criterion to determine the feasibility of a scheduling strategy.

III. System Model and Problem Formulation

In this section, we describe the system model and formulate the tree-based data gathering into an optimization problem.

Consider a sensor network that consists of a set of static sensor nodes, denoted as N. One of the sensor nodes in the network acts as a sink node to collect and process data from other sensor nodes. Each sensor node can select a transmitting power level from set $P = \{p_i | 0 \leq p_i \leq p_{max}\}$ for wireless communications. To ensure high reliability, the SINR at the receiver of a link should be greater than a minimum threshold λ. For any two sensor nodes i and j, a directed link from i to j exists if the SINR at node j is greater than λ, when node i transmits at power level p_{max} while there is no interference from other links. Note that link (i, j) is directed since the interference and noise levels at i and j are asymmetric. We use set L to denote all the directed links in the network.

All sensor nodes in the network access the wireless medium using a TDMA-based MAC mechanism, where MAC time is divided into fixed-length time slots. In each time slot, a packet can be transmitted on a link that is scheduled to be active. Sensor nodes sense their nearby field periodically and all nodes have the same sensing frequency, which implies that in a given period, all sensor nodes have the same amount of data to be sent to the sink. We define a *time frame* as a set of continuous time slots, in which the sensed data from each node is collected once. Then it is reasonable to assume that in a time frame, each node has one packet to transmit to the sink and the packets of all nodes are of the same length. On the other hand, the sink node needs to receive the data packets from all sensor nodes in each time frame. For any node i, we define its *traffic demand* D_i as the number of packets it generates in a time frame. Then $D_i = 1$ for a regular sensor node. For the sink node, it does not generate any packet but "consumes" the data packets from all other $|N| - 1$ sensor nodes. The traffic demand of the sink node is thus defined as $-(|N| - 1)$ so that the flow conservation property of the network is maintained. Formally, the traffic demand of node i

$$D_i = \begin{cases} 1, i \neq \text{sink} \\ 1 - |N|, i = \text{sink} \end{cases}$$

A. Data Gathering and Link Scheduling Constraint

We select a subset of links from set L to form a data gathering tree T rooted at the sink, such that each node forwards its sensed data to the sink through the directed path between them. An example data gathering tree is illustrated in Fig. 1. For link (i, j) in L, we use a binary variable $x_{i,j}$ to denote whether it is on the data gathering tree

$$x_{i,j} = \begin{cases} 1, (i, j) \in T \\ 0, \text{otherwise} \end{cases}$$

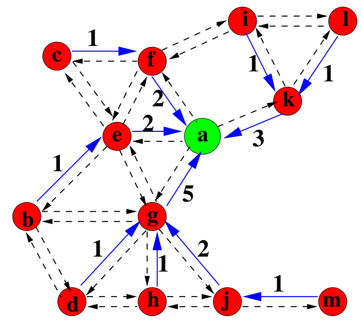

Fig. 1. An example of data gathering tree in a WSN, where each node has a packet to send to the sink. In the network, node a is the sink node while other nodes are regular sensor nodes. the blue arcs denote the links on the data gathering tree, while the black dotted arcs represent other directed links in the network. The numbers denote the traffic load on each link.

In addition, we define K as a large number of time slots that is sufficient to deliver the packets from all nodes to the sink, even when only one link is active for transmission in each time slot. We use a binary variable $s_{i,j}^t$ to indicate whether link (i, j) is scheduled to transmit in time slot t. If $s_{i,j}^t = 1$, we say link (i, j) is an *active link* for transmission in time slot t. We have

$$s_{i,j}^t = \begin{cases} 1, (i, j) \text{ is active in time slot } t \\ 0, \text{otherwise.} \end{cases} \quad 1 \leq t \leq K \qquad (1)$$

Clearly, a link should not be scheduled to transmit if it is not on the data gathering tree. This constraint can be stated as

$$s_{i,j}^t \leq x_{i,j}, 1 \leq t \leq K$$

In addition, due to the half duplex characteristics of transceivers equipped by sensor nodes, a sensor node can only transmit or receive in a time slot. In other words, for a node, at most one incoming link or outgoing link can be active at any time. This constraint can be written as

$$\sum_{i \in N, i \neq j} s_{i,j}^t + \sum_{m \in N, m \neq j} s_{j,m}^t \leq 1, \quad 1 \leq t \leq K$$

On the data gathering tree, every node except the sink needs to transmit its own packet and relay all incoming packets from its children to its parent. Note that in this paper, we do not consider data aggregation at intermediate nodes, such that our model will not be limited to only specific applications whose data is spatiotemporally correlated. Then a sensor node will need a separate time slot to relay every received packet. In other words, the number of active time slots for outgoing links of a node should be equal to its active time slots for incoming links plus one extra time slot, which is used to transmit its own data. This constraint can be formally expressed as

$$\sum_{t=1}^{K} \sum_{i \in N, i \neq j} s_{i,j}^t + D_j = \sum_{t=1}^{K} \sum_{m \in N, m \neq j} s_{j,m}^t$$

However, it should be mentioned that our model can be easily extended to support data aggregation by incorporating the aggregation ratio into the above flow conservation constraint.

B. Signal to Interference-Plus-Noise Ratio Constraint

As discussed earlier, the SINR of active links on the data gathering tree should be greater than threshold λ to provide reliable transmissions. A determinant factor affecting the SINR of a link is the received signal strength level, which is related to the transmitting power level, transmission distance and the signal propagation model. We adopt the log-distance path loss propagation model [7] to estimate SINR in this paper, as it can describe the shadowing fading precisely. Let $r_{i,j}^{dBm}$ denote the received power at node j in decibel for link (i,j). The received power $r_{i,j}^{dBm}$ can be derived as follows

$$r_{i,j}^{dBm} = p_i^{dBm} - PL_{d_0}^{dB} - 10\rho \log \frac{d_{i,j}}{d_0} + R_\delta^{dB}$$

where p_i^{dBm} is the transmitting power level of node i in decibel, d_0 is a constant value used as a reference distance by the model, $PL_{d_0}^{dB}$ is the path loss at d_0 in decibel, ρ is the path loss exponent, $d_{i,j}$ is the distance between nodes i and j, and R_δ^{dB} is a zero-mean Gaussian random variable with standard deviation δ.

The received signal strength of link (i,j) can also be expressed in non-logarithmic form, in which the received power is the product of the channel gain $g(i,j)$ from i to j and the transmitting power p_i of node j in watt, which is given by

$$r_{i,j} = g(i,j) \cdot p_i = \left(\frac{d_0^\rho}{d_{i,j}^\rho} \cdot 10^{-\frac{PL_{d_0}^{dB}}{10} + \frac{R_\delta^{dB}}{10}} \right) \cdot p_i$$

In the definition of link SINR, interference is the sum of received power from all unintended transmitters while noise refers to background noise. Since a different subset of links on the data gathering tree will be scheduled in each time slot, the SINR of link (i,j) is different in each time slot due to the change of transmitting power and interference sources. In time slot t, interference comes only from the transmitters of links that are scheduled to be active in that time slot. Clearly, the SINR of link (i,j) is meaningful only if the link is scheduled. Thus we define the SINR of link (i,j) in time slot t as zero when $s_{i,j}^t = 0$. Moreover, any node except i will be a source of interference to link (i,j) if one of its outgoing links is scheduled in the same time slot. We use $q_{i,j}^t$ to denote the SINR of link (i,j) in time slot t, which equals the received power from i to j divided by the sum of received power at j from all other active transmitters plus noise. $q_{i,j}^t$ can be formally defined as

$$q_{i,j}^t = \frac{s_{i,j}^t \cdot r_{i,j}}{\sum\limits_{m \in N, m \neq i} (\sum\limits_{n \in N, n \neq j} s_{m,n}^t) \cdot r_{m,j} + B_j}$$

where B_j is the background noise sensed at node j.

Finally, if link (i,j) is scheduled in time slot t, then its SINR should be greater than threshold λ.

$$q_{i,j}^t \geq s_{i,j}^t \cdot \lambda$$

C. Problem Formulation

The SINR-based data gathering problem in a WSN can be formally described as follows. Given a wireless sensor network consisting of a set of sensor nodes, N, and a set of directed links among them, L, find a subset of links from L to form a tree T rooted at the sink node, allocate transmitting time slots for links on T, and assign transmitting power to active links in each time slot, such that the traffic demand of all nodes is satisfied, and the number of allocated time slots is minimized. Thus, the reliability of data gathering can be guaranteed, since the SINR of all active links is above the threshold λ.

The SINR-based data gathering problem can be formulated into an optimization problem, with the objective to minimizing the number of allocated time slots, which can be expressed as

$$\sum_{t=1}^{K} \max_{(i,j) \in L} s_{i,j}^t$$

As defined in Equation (1), $s_{i,j}^t = 1$ if link (i,j) is active in time slot t. Then $\max_{(i,j) \in L} s_{i,j}^t = 1$ if at least one link is active in t, indicating that time slot t is allocated. Accordingly, $\sum_{t=1}^{K} \max_{(i,j) \in L} s_{i,j}^t$ is the overall number of allocated time slots. The formulated optimization problem is given as follows

Minimize

$$\sum_{t=1}^{K} \max_{(i,j) \in L} s_{i,j}^t$$

Subject to

$$0 \leq p_i \leq p_{max}, \qquad \forall i \in N \tag{2}$$

$$D_i = \begin{cases} 1, i \neq \text{sink} \\ 1 - |N|, i = \text{sink} \end{cases} \qquad \forall i \in N \tag{3}$$

$$1 \leq t \leq K \tag{4}$$

$$x_{i,j} = \begin{cases} 1, \ (i,j) \in T \\ 0, \ \text{otherwise.} \end{cases} \qquad \forall (i,j) \in L \tag{5}$$

$$s_{i,j}^t \leq x_{i,j}, \forall (i,j) \in L, \qquad 1 \leq t \leq K \tag{6}$$

$$\sum_{i \in N, i \neq j} s_{i,j}^t + \sum_{m \in N, m \neq j} s_{j,m}^t \leq 1, \qquad 1 \leq t \leq K \tag{7}$$

$$\sum_{t=1}^{K} \sum_{i \in N, i \neq j} s_{i,j}^t + D_j = \sum_{t=1}^{K} \sum_{m \in N, m \neq j} s_{j,m}^t, \quad \forall j \in N \tag{8}$$

$$r_{i,j} = \left(\frac{d_0^\rho}{d_{i,j}^\rho} \cdot 10^{-\frac{PL_{d_0}^{dB}}{10} + \frac{R_\delta^{dB}}{10}} \right) \cdot p_i, \qquad \forall (i,j) \in L \tag{9}$$

$$q_{i,j}^t = \frac{s_{i,j}^t \cdot r_{i,j}}{\sum\limits_{m \in N, m \neq i} (\sum\limits_{n \in N, n \neq j} s_{m,n}^t) \cdot r_{m,j} + B_j}, \ \forall (i,j) \in L \tag{10}$$

$$q_{i,j}^t \geq s_{i,j}^t \cdot \lambda, \qquad \forall (i,j) \in L, 1 \leq t \leq K \tag{11}$$

In the above formulation, Equations (2) and (3) specify the range of transmitting power and the traffic demands of all nodes, respectively; Equations (6), (7), (8) and (11) are data gathering, link scheduling and SINR threshold constraints; Equations (9) and (10) are used to determine the received power

TABLE I
LIST OF NOTATIONS USED IN PROBLEM FORMULATION

N	Set of sensor nodes
L	Set of directed links
P	Set of transmitting power levels
D	Set of traffic demands
T	Data gathering tree
K	A large number of time slots for one round of data gathering
λ	Minimum SINR threshold
ρ	Path loss exponent for the propagation model
p_{max}	Maximum transmitting power of all nodes
p_i	Transmitting power of node i
D_i	Traffic demand to the sink from node i
$x_{i,j}$	Indicator whether (i,j) is on data gathering tree T
$s^t_{i,j}$	Indicator whether (i,j) is scheduled in time slot t
$r_{i,j}$	Received power at node j from node i
$d_{i,j}$	Distance between node i and node j
$q^t_{i,j}$	SINR of link (i,j) in time slot t
$g(i,j)$	Channel gain from node i to node j
B_i	Background noise at node i

level and the SINR of each link, respectively. The notations used in the formulation is given in Table I. After the optimal results are derived, the time slots that have no active link are removed and the remaining time slots form a time frame.

D. NP-Hardness

We have the following lemma concerning the NP-hardness of the above optimization problem.

Lemma 1: The SINR-based data gathering problem in a WSN is NP-hard.

Proof: We prove the lemma by reducing from the *max-connections* problem, which has been proved to be NP-hard in [24]. The max-connection problem is to maximize the number of links that satisfy the minimum SINR threshold in an arbitrary wireless network, by choosing the transmitting power of all links. The SINR-based data gathering problem can be expressed in the following form: Among all the spanning trees in the network, find the tree that leads to the least data gathering time while all links satisfy the SINR constraint.

In a spanning tree, the *traffic load* on each link for data gathering without data aggregation is deterministic. For any link (i,j) on T, we define its traffic load as the sum of the traffic demand D_i of node i and the traffic demands of all descendants of node i. The problem of finding the minimum data gathering time on a spanning tree can be further decomposed into a series of max-connection subproblems. In each max-connection subproblem, an auxiliary graph is constructed from the data gathering tree. An edge is added to the auxiliary graph if the corresponding link on the data gathering tree has remaining traffic load. After resolving the max-connection subproblem on the auxiliary graph, the remaining traffic load of the data gathering tree is updated. This step is repeated until the traffic load of every link equals zero. Then the number of iterations is the data gathering latency of the tree. Specifically, in a WSN where all sensor nodes are placed on a line, there exists only one spanning tree in the network and it is the same as the network topology. Then the SINR-based optimization problem is essentially the combination of several max-connection subproblems. Therefore, the SINR-based optimization problem is NP-hard. ■

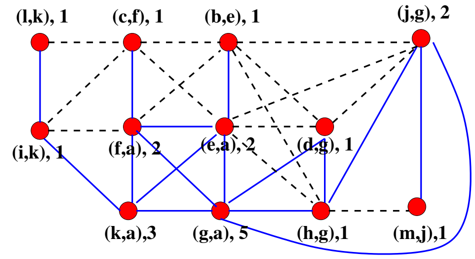

Fig. 2. Enhanced power-based interference graph for the data gathering tree in Fig. 1, where each vertex is associated with a link on the tree, and the number is the traffic load on the link. A black dotted edge between two vertices means that the two associated links share a common sensor node, while a blue edge means that the two associated links cannot be scheduled in the same time slot.

IV. HEURISTIC ALGORITHMS

In the above formulated optimization problem, the number of constraints increases exponentially along with the number of sensor nodes, making it prohibitive to derive optimal solutions through mathematical tools. To make the problem tractable, we divide the problem into two subproblems: (1) Construct a low-latency data gathering tree; (2) Schedule links and assign transmitting power to active links in each time slot, such that the data gathering time is minimized while the SINR of all active links is greater than λ. Next, we first introduce an enhanced power-based interference graph to facilitate solving these two subproblems, then we present a heuristic algorithm for each subproblem.

A. Enhanced Power-Based Interference Graph

As discussed in Section II, a power-based interference graph was introduced in [20], in which two vertices are connected if it is impossible to schedule the two corresponding links in the WSN simultaneously given any power assignment. This graph can be useful when scheduling links on the data gathering tree. However, it does not reflect the traffic load of links, which will be needed in our heuristic algorithms. Thus we extend the graph into an *enhanced power-based interference graph* for data gathering, defined as a weighted undirected graph $I = (V, E, W)$, where V is the set of vertices, E is the set of edges to denote interference relationships, and W is the set of weights for all vertices. Each vertex in V is associated with a link on the data gathering tree. Two vertices in V are connected if both associated links have positive traffic load, while there is no feasible power assignment for them to satisfy the minimum SINR threshold or they share a common node. We say two links are *incompatible* if their associated vertices on I are connected. The vertex weight is defined as the remaining traffic load of the associated link. The enhanced power-based interference graph for the data gathering tree in Fig. 1 is shown in Fig. 2.

B. Construction of Data Gathering Tree

For tree-based data gathering, the number of required time slots depends on the tree topology. In [11], a lower bound on data gathering time, $\max\{2n_\delta - 1, |N|\}$, was given based on the assumption that all interference can be totally eliminated, where n_δ is the maximum subtree size of all subtrees rooted at the sink on the data gathering tree. Intuitively, n_δ is relatively small if sensor nodes are distributed in multiple subtrees in a balanced manner. On the other hand, as it is impossible to schedule two incompatible links in the same time slot, we should avoid adding links that are incompatible with many tree links when constructing the data gathering tree, so as to reduce the number of required time slots. Based on these observations, we devise an algorithm, named *low latency high compatibility* (LLHC) algorithm, to construct the data gathering tree in WSNs.

In LLHC, a *height* is assigned to each node by traversing the network using the breadth first search (BFS) algorithm. The traverse begins at the sink node such that the height of a node is its shortest distance to the sink in hops. The terms height and level will be used interchangeably in the rest of the paper. In addition, an enhanced power-based interference graph I is constructed for the network. The data gathering tree T is initially empty. At first, all nodes at level 1 and their directed links to the sink node are added to T. Then each level-1 node is selected as the root of a subtree. As a result, the number of subtrees is maximized due to the nature of BFS. All other sensor nodes are added to T in an ascending order of their levels. A node at level m can choose any node at level $m-1$ as a parent candidate as long as there exists a directed link between them in the network. As discussed earlier, the candidate that minimizes the maximum subtree size and introduces a new link that is compatible with most links on T should be selected as the parent. A weight is defined for parent candidates to reflect these two factors. For parent candidate j of node i, the weight is defined as the sum of the size of the subtree that node j belongs to and the number of links on T that link (i, j) is incompatible with. Candidate j' that has the smallest weight is selected as the parent of node i and link (i, j') is added to T. This procedure is repeated until all nodes are included in T.

The algorithm is guaranteed to terminate since every time a new node is connected to the data gathering tree and the number of nodes is finite. Table II gives the pseudo code of LLHC. From the table, we can see that it takes $O(|L|^2)$ time to generate the enhanced power-based interference graph and $O(|N| + |L|)$ to traverse all nodes using BFS. Assume that there are M levels of nodes and the number of nodes in level m is n_m, such that $n_1 + n_2 + \cdots + n_m = |N|$. In the worst case, any node in level $m-1$ can be a parent candidate for each node at level m. Then it takes $O(n_{m-1} \cdot n_m)$ to connect nodes at level m into T, by assuming the time to determine the weight for a parent candidate is constant. Then the time to connect all nodes is $O(n_1 \cdot n_2 + n_2 \cdot n_3 + \cdots + n_{M-1} \cdot n_M) = O(|N|^2)$. Thus the overall time complexity of LLHC algorithm is $O(|L|^2 + (|N| + |L|) + |N|^2) = O(|L|^2)$.

C. Link Scheduling and Power Assignment

After obtaining a data gathering tree by LLHC algorithm, the next step is to allocate time slots and transmitting power to links on the tree to minimize data gathering latency. When scheduling links for a new time slot, links with high traffic load should be considered first as they need more time slots to transmit the traffic than other links. On the other hand, a link whose associated vertex on I has a large degree should

TABLE II
LOW LATENCY HIGH COMPATIBILITY (LLHC) ALGORITHM

Input:
  Set of sensor nodes N;
  Set of directed links L;
Output:
  Data Gathering Tree T;
Algorithm:
  Initialize transmitting power to p_{max} for all nodes in N;
  Initialize $parent$ to -1 for all nodes in N;
  Generate enhanced power interference matrix I;
  Get $level(i)$ for all nodes by traversing the network using BFS;
  Let $level_{max}$ be the maximum level of all nodes;
  $T = \emptyset$;
  for each node n at level 1
    $parent(n) = sink$, $subtree(n) = n$;
    $numsub(n) = 1$, $T = T \bigcup (n, sink)$;
  end for
  for each level m from 2 to $level_{max}$
    for each node n at level m
      $C = \{c \in N |\ level(c) = m-1, parent(c) \neq -1, (c, n) \in L\}$
      for each c in C
        $count(c) = numsub(subtree(c))$;
        for each link f on T
          if f is connected with (c, n) on I
            $int(c)$++;
          end if
        end for
        $weight(c) = int(c) + count(c)$;
      end for
      Let c' be the node in C that has the minimum weight;
      $parent(n) = c'$;
      $subtree(n) = subtree(c')$;
      $numsub(subtree(n)) = numsub(subtree(n)) + 1$;
      $T = T \bigcup (n, c')$;
    end for
  end for

be given high priority also, since it prevents the scheduling of all other links incompatible with it in the same time slot. Based on these observations, we propose a *maximum weight first* (MWF) algorithm for link scheduling and transmitting power assignment, where the weight of a link is defined as the sum of the remaining traffic load over the link and the degree of the vertex that the link is associated with on I.

Given a set of links, it is challenging to determine whether there exists a power assignment vector for their transmitters, such that the SINR constraints of all links are satisfied. Such a transmitting power assignment may not exist even if the associated vertices on the interference graph for these links are independent of each other, due to the accumulative nature of interference. The brutal force search method is infeasible when the number of links is large. For a set of links, M, that do not share a common node, we define an $|M| \times |M|$ feasibility matrix F as follows

$$F = \begin{bmatrix} 0 & \frac{g(2_t,1_r)}{g(1_t,1_r)} & \frac{g(3_t,1_r)}{g(1_t,1_r)} & \frac{g(m_t,1_r)}{g(1_t,1_r)} \\ \frac{g(1_t,2_r)}{g(2_t,2_r)} & 0 & \frac{g(3_t,2_r)}{g(2_t,2_r)} & \frac{g(m_t,2_r)}{g(2_t,2_r)} \\ \vdots & \vdots & \vdots & \vdots \\ \frac{g(1_t,m_r)}{g(m_t,m_r)} & \frac{g(2_t,m_r)}{g(m_t,m_r)} & \frac{g(m-1_t,m_r)}{g(m_t,m_r)} & 0 \end{bmatrix}$$

where the transmitter and receiver of link m is denoted by m_t and m_r, respectively. The $(i, j)^{th}$ element of feasible matrix F

TABLE III
LINK SCHEDULING FOR DATA GATHERING TREE IN FIG. 1.

time slot	active links		time slot	active links	
1	g→a	c→f	9	k→a	
2	g→a	l→k	10	g→a	
3	g→a	m→j	11	k→a	
4	e→a		12	i→k	j→g
5	g→a		13	b→e	
6	e→a		14	k→a	
7	f→a	d→g	15	j→g	
8	f→a	h→g			

is given by

$$F(i,j) = \begin{cases} 0, & i = j \\ \frac{g(j_t, i_r)}{g(i_t, i_r)} & i \neq j \end{cases}$$

where $g(j_t, i_r)$ is the channel gain from the transmitter of link j to the receiver of link i, and $g(i_t, i_r)$ is the channel gain of link i.

When noise at receivers is ignored, all links in M can be scheduled simultaneously if and only if the reciprocal of the Perron eigenvalue ρ of matrix F is greater than the SINR threshold λ, where any positive multiple of the corresponding Perron eigenvector can be used as the power assignment vector p for all links. The Perron eigenvalue of a nonnegative square matrix is its largest positive eigenvalue, while the Perron eigenvector is the corresponding eigenvector of Perron eigenvalue.

When noise at receivers is considered, a constant factor c needs to be multiplied to the above derived power assignment vector so as to ensure the SINR constraints are fulfilled. c should satisfy the following condition

$$c \geq \max_{m \in M} \left\{ \frac{B_{m_r}}{g(m_t, m_r)(\delta^{-1} - \rho(F)) p_{m_t}} \right\}$$

where B_{m_r} is the background noise at the receiver of link m, and p_{m_t} is the transmitting power of link m. The necessity and sufficiency of this condition was proved in detail in [23]. We use this condition in MWF to determine whether a set of links can be scheduled in the same time slot.

In MWF algorithm, the enhanced power-based interference graph I for data gathering tree T is first constructed. Then all links on T are sorted in a descending order of their weight. A new time slot t is allocated and link m that has the highest weight is put into the set of active links for time slot t, S_t. The traffic load on link m is decreased by 1. Then the compatibility between link m' that has the next highest weight and all links in S_t is examined. If link m' is compatible with all of them, a feasibility matrix F is constructed for links $S_t \bigcup m'$. If a feasible power assignment can be found for F, link m' is added to S_t and the traffic load on m' is decreased by 1. Otherwise, the next link on the sorted list is examined. The active link set S_t for time slot t is finalized after all links that have remaining traffic loads are examined once. After that, interference graph I is updated accordingly. A new time slot $t+1$ is allocated and this procedure is repeated until the traffic loads of all links on T become zero. As an example, the link scheduling generated by MWF algorithm for the data gathering tree in Fig. 1 is given in Table III.

The algorithm is guaranteed to terminate since in each time slot at least one link is scheduled and the overall traffic load is finite. The pseudo code of MWF is described in Table IV. To analyze the time complexity of MWF, we assume that u links are scheduled in each time slot on average. In the worst case, the first u links in the sorted link set are always selected into the active link set. Then for each of the remaining $|T| - u$ links, it takes $O(u)$ time to check whether the link is compatible with the first u links. If so, it takes another $O((u+1)^3)$ time to verify the feasibility of their matrix F, as the time complexity of calculating eigenvalue and eigenvector of an $n \times n$ matrix is $O(n^3)$. Otherwise, the link cannot be added to the active link set and the compatibility of the next remaining link is checked. In the worst scenario, all sensors are scattered on a line and the sink is at one end of the line, then the data gathering tree also has a line topology. The summed traffic load of this scenario is $1 + 2 + \cdots + |T| = |T|(|T|+1)/2$. Consequently, the number of required time slots is $|T|(|T|+1)/(2u)$. The overall time complexity of MWF can then be approximated as

$$O(|T| * (u + (u+1)^3) \cdot (|T|(|T|+1)/(2u))) \approx O(|T|^3 \cdot u^2)$$

Since at most one half of links on the data gathering tree can be scheduled at any time slot, u is bounded by $|T|$, thus the time complexity of MWF is $O(|T|^5)$.

In practical WSNs, every sensor node can broadcast a fixed-power beacon message at a low frequency, and other nodes determine the channel gain of links from this node by measuring the received power level of the beacon message. Such channel gain information is further transmitted to the sink node. The sink node then executes both LLHC and MWF algorithms and disseminates the derived tree construction, link scheduling and power assignment outputs to all nodes. Note that LLHC and MWF algorithms only need to be executed periodically

TABLE IV
MAXIMUM WEIGHT FIRST (MWF) ALGORITHM

```
Input:
  Data Gathering Tree T;
  Set of transmitting power levels P;
  Set of traffic demands D;
Output:
  Link Scheduling Matrix S and Power Assignment Matrix U;
Algorithm:
  Generate enhanced power-based interference graph I for T;
  Determine the traffic load on each link in T;
  for each m ∈ T
    Define degree(m) of link m as the degree
    of its associated vertex on I;
    weight(m) = load(m) + degree(m);
  end for
  t = 1;
  sumd = sum(load);
  while (sumd > 0)
    Sort links on T in the descending order of their weight into T';
    for each m in T'
      if m does not conflict with any link in S_t
      and there exists feasible power assignment for S_t ⋃ m
        S_t = S_t ⋃ m;
        load(m) − −;
      end if
    end for
    Assign transmitting power vector U_t for links in S_t;
    Update enhanced power-based interference graph I;
    Update weight for all nodes;
    t + +;
end while
```

to reflect variance of channel gain or change of network topology due to depleted-battery nodes. Thus the overhead of the proposed algorithms is negligible, as the WSN spends most of time and energy in gathering sensed data.

V. Performance Evaluations

In this section, we evaluate the performance of the LLHC and MWF algorithms for data gathering through simulations. We first study the effectiveness of the proposed algorithms under various node densities and SINR thresholds. We then examine the distribution of data gathering load in terms of average relaying traffic and the maximum buffer length of sensor nodes. After that, we explore the impact of the sink location on data gathering latency. For comparison, the breadth first search (BFS) algorithm and maximizing lifetime tree (MLT) algorithm [4] are implemented as well to generate data gathering trees, while the integrated scheduling and power control (ISPA) algorithm [20] and the increasing demand greedy scheduling (IDGS) algorithm [22] are implemented to schedule links on the data gathering tree.

In the simulations, sensor nodes are randomly scattered in a square field of $500 \times 500m^2$. The maximum transmitting power of all nodes is set to $-10dBm$. Moreover, all nodes communicate over a $5MHz$ frequency band and the mean value of the additive white Gaussian noise (AWGN) is $-97dBm$. The log-distance path loss model is used as the propagation model, where the pass loss exponent ρ is 3 and the standard deviation of X_δ is $7dB$. If not otherwise specified, the minimum SINR threshold is set to $10dB$ while the sink node is deployed at the center of the field. All results are averaged over 100 runs.

We first evaluate the proposed LLHC and MWF algorithms in terms of the number of required time slots for data gathering. The number of sensor nodes varies from 50 to 200 in a step of 25. The simulation results are plotted in Fig. 3(a), where LLHC and MWF are performed jointly, while trees generated by BFS and MLT are scheduled by ISPA and IDGS link scheduling algorithms separately. It is notable that the proposed algorithms always need less data gathering time than any combination of the compared algorithms. Moreover, the advantage of LLHC and MWF is more obvious when the node density is relatively high. For example, when 200 nodes are deployed in the field, the result of the proposed algorithms is 13% and 28% better than the results derived from MLT-IDGS and MLT-ISPA, respectively. It verifies that our tree construction and link scheduling strategies are more capable of reducing data gathering latency while guaranteeing high reliability. Note that given the same data gathering tree, IDGS takes fewer time slots for data gathering than ISPA, as traffic load on links is considered in IDGS. On the other hand, it is difficult to see whether BFS or MLT is better in shortening data gathering latency, since different link scheduling algorithms lead to different results.

We now examine the number of required time slots for the proposed algorithms under different SINR threshold constraints. The number of sensor nodes is fixed to 100 while the minimum SINR threshold is varied from $0dB$ to $30dB$

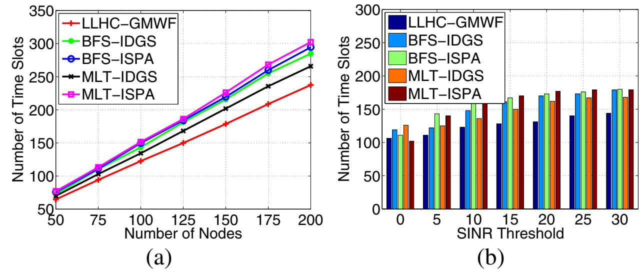

Fig. 3. Number of required time slots for data gathering under different node densities and minimum SINR thresholds.

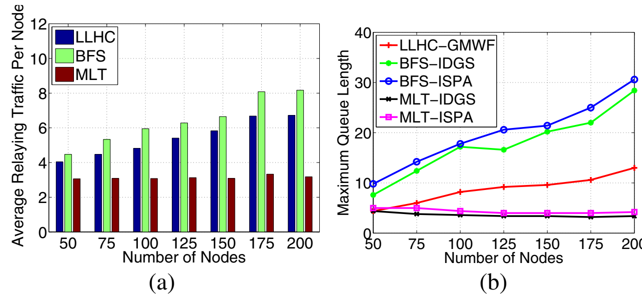

Fig. 4. Average relaying traffic and maximum queue length on each node for different data gathering strategies.

in a step of $5dB$. The simulation results of LLHC-MWF and four combinations of the compared algorithms are shown in Fig. 3(b). Clearly, the data gathering time for all algorithms increases when the minimum SINR threshold grows, since fewer links can be scheduled in each time slot when links are more sensitive to interference. Nevertheless, the proposed algorithms always outperform the compared algorithms under different minimum SINR constraints. Moreover, the benefits of LLHC and MWF are more remarkable when the minimum SINR threshold is relatively strict. Specifically, when the SINR threshold is $20dB$, the proposed algorithms take 31 and 46 fewer time slots for data gathering than MLT-IDGS and BFS-ISPA, respectively. We can also observe that the advantage of LLHC-MWF over other algorithms drops slightly when the SINR threshold grows beyond $20dB$. The reason is that a large number of links on the tree become incompatible with each other when the SINR threshold is high and thus cannot be scheduled in the same time slot, even though they experience less interference than other links that are not selected into the tree.

Next, we study the average traffic a node has to relay on data gathering trees generated by various algorithms. The results are plotted in Fig. 4(a), where the number of nodes varies from 50 to 200. We can see that the average relaying traffic for LLHC is lower than that of BFS, while higher than that of MLT. The reason is that different from BFS, LLHC algorithm takes the subtree size into consideration when constructing trees such that the traffic load is more uniformly distributed over the network. The average relaying traffic of MLT remains almost unchanged as the node density grows, since it tries to minimize the maximum traffic load of links at every level of the tree. However, as shown in Fig. 3, it takes much longer time than LLHC to finish one round of data gathering, indicating a tradeoff between relaying load and data gathering latency.

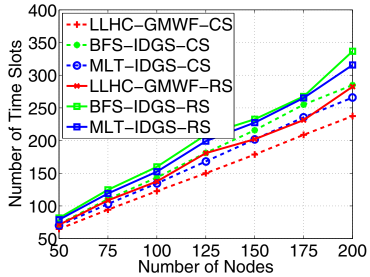

Fig. 5. Number of required time slots for data gathering in WSNs where the sink is randomly deployed or located at the center of the field.

We also examine the maximum queue length on every node during data gathering, so as to determine the minimum buffer size for sensors. The simulation results are shown in Fig. 4(b). It is notable that the maximum queue length of LLHC and MWF is much shorter than that of BFS-based data gathering strategies. In addition, the maximum queue length of the proposed algorithms increases slowly as the node density grows. When there are 200 nodes in the field, the maximum queue length of LLHC-MWF is 13, which is only about one third of the BFS-based strategies. Suppose the packet size is $1KB$, then the buffer requirement of the proposed algorithms is less than $20KB$, which is affordable by most off-the-shelf sensors. The maximum queue length of MLT-based data gathering is less than 5 regardless of the node density, which can be explained similarly to the average relaying traffic per node discussed earlier. We can also observe that IDGS needs less buffer than ISPA over the same data gathering tree. This is because that more links are scheduled by IDGS in each time slot than by ISPA on average, as implied in Fig. 3.

Finally, we study the impact of the sink location on the data gathering latency. Two sets of experiments are conducted. The sink node is deployed randomly in one set while located at the center of the field in the other set. The results are plotted in Fig. 5, where CS and RS denote the centrally-located sink and randomly-located sink, respectively. It is clear that more time is needed by all data gathering strategies if the sink is randomly located. The reason is that if the sink is at the margin of the network, then few nodes are capable of transmitting data to the sink directly, making them the bottleneck of data gathering. However, the proposed algorithms can still reduce the data gathering latency remarkably compared with other algorithms.

VI. Conclusions

In this paper, we have studied tree-based data gathering in WSNs. Our objective is to gather data from all sensors with low latency and high reliability, by carefully constructing a data gathering tree, scheduling links on the tree, and assigning transmitting power levels to active links in each time slot. We formulated the problem into an optimization problem and proved it is NP-hard. We then divided the problem into two subproblems and provided a heuristic algorithm to each of them. We have conducted extensive simulations and the results demonstrate that the propose algorithms can significantly reduce the data gathering latency compared to other schemes under different node densities and minimum SINR thresholds, regardless of the location of the sink node. The results also show that the traffic load is more balanced throughout the network in our schemes, and thus the network lifetime is prolonged as well.

References

[1] N. Thepvilojanapong, Y. Tobe and K. Sezaki, "On the construction of efficient data gathering tree in wireless sensor networks," *IEEE ISCAS*, pp. 648-651, May 2005.

[2] C. Liu, K. Wu and J. Pei, "An energy-efficient data collection framework for wireless sensor networks by exploiting spatiotemporal correlation," *IEEE Trans. Parallel and Distributed Systems*, pp. 1010-1023, Jul. 2007.

[3] Y. Wu, S. Fahmy and N. B. Shroff, "On the construction of a maximum-lifetime data gathering tree in sensor networks: NP-completeness and approximation algorithm," *IEEE INFOCOM*, pp. 356-360, April 2008.

[4] D. Luo, X. Zhu, X. Wu and G. Chen, "Maximizing lifetime for the shortest path aggregation tree in wireless sensor networks," *IEEE INFOCOM*, pp. 1566-1574, April 2011.

[5] M. Ma and Y. Yang, "SenCar: An energy efficient data gathering mechanism for large scale multihop sensor networks," *IEEE Trans. Parallel and Distributed Systems*, vol.18, no.10, pp. 1476-1488, Oct. 2007.

[6] J. Zhao and R. Govindan, "Understanding packet delivery performance in dense wireless sensor networks," *ACM SenSys*, pp. 1-13, 2003.

[7] M. Zuniga and B. Krishnamachari, "Analyzing the transitional region in low power wireless links," *IEEE SECON*, pp. 517-526, Oct. 2004.

[8] H. Choi, J. Wang and E. A. Hughes, "Scheduling for information gathering on sensor network," *Wireless Networks*, vol. 15, pp. 127-140, Jan. 2009.

[9] W. C. Cheng, C. Chou, L. Golubchik, S. Khuller and Y. C. Wan, "A coordinated data collection approach: design, evaluation, and comparison," *IEEE JSAC*, pp. 2004-2018, Dec. 2004.

[10] S. Gandham, Y. Zhang and Q. Huang, "Distributed time-optimal scheduling for convergecast in wireless sensor networks," *Computer Networks*, vol. 52, Feb. 2008.

[11] O. Durmaz Incel, A. Ghosh, B. Krishnamachari and K. Chintalapudi, "Fast data collection in tree-based wireless sensor networks," *IEEE Trans. Mobile Computing*, vol. 11, no. 1, pp. 86-99, Jan. 2012.

[12] R. Maheshwari, S. Jain and S.R. Das, "A measurement study of interference modeling and scheduling in low-power wireless networks," *ACM SenSys*, pp. 141-154, 2008.

[13] O. Goussevskaia, Y.A. Oswald and R. Wattenhofer, "Complexity in geometric SINR," *ACM MobiHoc*, pp. 100-109, September 2007.

[14] S. Kompella, J. E. Wieselthier and A. Ephremides, "A cross-layer approach to optimal wireless link scheduling with SINR constraints," *IEEE Military Communications Conference*, pp. 1-7, Oct. 2007.

[15] T. Moscibroda, R. Wattenhofer and A. Zollinger, "Topology control meets SINR: the scheduling complexity of arbitrary topologies," *ACM MobiHoc*, pp. 310-321, May 2006.

[16] D. M. Blough, G. Resta and P. Santi, "Approximation algorithms for wireless link scheduling with SINR-based interference," *IEEE/ACM Trans. Networking*, vol. 18, pp. 1701-1712, Dec. 2010.

[17] F. Yuan, B. Zhang and C. Yang, "A joint power control and link scheduling strategy for consecutive transmission in TDMA wireless networks," *IEEE PIMRC* , pp. 2477-2482, Sept. 2010.

[18] P. Wan, O. Frieder, X. Jia, F. Yao, X. Xu and S. Tang, "Wireless link scheduling under physical interference model," *IEEE INFOCOM*, pp. 838-845, April 2011.

[19] S. Sengupta, M. Chatterjee and K. A. Kwiat, "A game theoretic framework for power control in wireless sensor networks," *IEEE Trans. Computers*, vol. 59, pp. 231-242, Feb. 2010.

[20] A. Behzad and I. Rubin, "Optimum integrated link scheduling and power control for multihop wireless networks," *IEEE Trans. Vehicular Technology*, pp. 194-205, Jan. 2007.

[21] S. A. Borbash and A. Ephremides, "Wireless link scheduling with power control and SINR constraints," *IEEE Trans. Information Theory*, pp. 5106-5111, Nov. 2006.

[22] L. Fu, C. Liew and J. Huang, "Joint power control and link scheduling in wireless networks for throughput optimization," *IEEE ICC*, pp. 3066-3072, May 2008.

[23] S. A. Borbash and A. Ephremides, "The feasibility of matchings in a wireless network," *IEEE Trans. Information Theory*, vol. 52, pp. 2749-2755, Nov. 2006.

[24] M. Andrews and M. Dinitz, "Maximizing capacity in arbitrary wireless networks in the SINR model: complexity and game theory," *IEEE INFOCOM*, pp. 1332-1340, April 2009.