SCHEDULING CONNECTED TOPOLOGIES IN WIRELESS SENSOR NETWORKS UNDER THE PHYSICAL INTERFERENCE MODEL

Welber W. F. dos Reis, Helder R. O. Rocha, Rodolfo S. Villaça, Renato E. N. Moraes
Departamento de Computação e Eletrônica – Universidade Federal do Espírito Santo
Rodovia BR 101 Norte, Km 60 – Bairro Litorâneo – 29932-540 – São Mateus – ES – Brazil
welber91@gmail.com, {helder.rocha, rodolfo.villaca, renato.moraes}@ufes.br

ABSTRACT

A fundamental aspect in performance engineering of wireless sensor networks is optimizing the set of links that can be concurrently activated to meet a given signal-to-interference-plus-noise ratio (SINR) threshold. The solution of this combinatorial problem is a key element in wireless link scheduling. Another key architectural goal in wireless sensor networks is connectivity. In this paper, we investigate the joint scheduling and connectivity problem in wireless sensor networks assuming the SINR model. We propose an algorithm to compute power efficient schedules with minimum number of time-slots needed to schedule all links that builds a connected communication graph such that the nodes can communicate without interference in the SINR model. The minimization of the schedule length has the effect of maximizing network throughput. Power efficient and interference free schedules reduce energy consumption. We report computational experiments to validate the effectiveness of the proposed algorithm.

KEYWORDS. Wireless Sensor Networks, shortest link schedule, heuristics.

1. Introduction

Wireless sensor networks (WSNs for short) are a particular type of ad hoc network, in which the nodes are "smart sensors", that is, small devices (approximately the size of a coin) equipped with advanced sensing functionalities (thermal, pressure, acoustic, are examples of such sensing abilities), a small processor, and a short-range wireless transceiver. This technology allows network nodes to communicate directly to each other, without the need for a fixed infrastructure (Santi, 2015). Applications of WSNs include battlefield surveillance, biological detection, home appliance, smart spaces and inventory tracking (Akyildiz, 2002).

There are many challenges related to WSN design. For instance, it is well known that energy is a precious resource in wireless networks due to the limited battery life. This is further aggravated in sensor networks since all nodes are limited in weight and size. In addition, reducing interference and increasing throughput on the medium access layer are one of the main goals in the sensor network design besides direct energy conservation by restriction of transmission power. Especially in sensor networks, the throughput and power efficiency of the wireless communication can be substantially determined by the quality of a link schedule (Katz et al., 2010). Wireless link scheduling algorithms can also be used to coordinate the transmissions of independent nodes in order to eliminate strong levels of interference.

Given a set of transmission link demands, a link schedule is a list of unicast transmissions sets such that all transmissions in the same set transmit concurrently. The length of the schedule is the time required to complete all the unicast transmissions sets, from when the first begins to when the last ends. The throughput of the schedule is proportional to the reciprocal of its length. A shortest schedule, or minimum schedule length, corresponds to the greatest possible throughput which can be achieved in the network (Borbash and Ephremides, 2006).

A schedule must satisfy a given link demands and obey some constraints. Among these constraints there is connectivity properties. Connectivity of sensor nodes is critical for WSNs, as data collected needs to be sent to sink or base stations (Xu et al., 2013). To permit a packet to be routed between any two nodes in a network, the corresponding communication graph must be connected. Establishing connectivity in a wireless network can be a complex task for which various (sometimes conflicting) objectives must be optimized. In addition to requiring connectivity, another constraints can be imposed on the network, including low power consumption (Moraes and Ribeiro, 2013), small average hop distance between sender-receiver pairs (Benkert et al., 2008), minimal interference (Blough et al., 2009), minimum number of relays (Cardoso and Moraes, 2014), capacity bounds (Halldórsson and Mitra, 2012), among others.

In this paper, we study the link scheduling problem for connectivity in wireless sensor networks subject to power and interference constraints. This problem consists in finding a set of power efficient links and scheduling them assuring interference free transmissions in order to achieve connectivity and minimum schedule length for a given placement of wireless nodes.

Concurrent communications can be separated in frequency by using multiple channels. If only a single channel is available, it is possible to divide time into frames, and then, frames can be divided into time slots, such that at each frame a node has the option to choose on which time slot to transmit. The link scheduling problem for connectivity can be defined more precisely as the task of minimize the total number of time-slots (minimum schedule length) needed to schedule all requests that form a connected network (communication graph) over a given set of n nodes located in the Euclidean plane. The resulting communication graph has reduced energy consumption and interference free transmissions.

The link scheduling problem for connectivity considered in this work was first abstracted and studied by (Moscibroda and Wattenhofer, 2006) in the context of SINR model. The autors showed how to compute in polynomial time an assignment of transmission powers with $O(\log^4 n)$ time slots to achieve strong connectivity of the resulting communication graph. Later, (Moscibroda, 2007) proposed a better algorithm using $O(\log^2 n)$ time slots, and (Halldórsson and Mitra, 2012) improved the bound to $O(\log n)$. (An et al., 2012) showed that this problem is NP-hard in the geometric SINR model and proposed two constant-factor approximation algorithms. These works studied the problem assuming arbitrary power levels, whereas (Avin et al., 2009) studied it on 1- and 2-dimensional networks with a uniform power level. Ignoring the background noise, they showed that the number of time-slots needed on the networks is constant.

These studies have been concerned with assigning minimum number of time-slots inducing a connected communication graph, some other researchers have focused on link scheduling problems for other applications such as data gathering (Gong and Yang, 2013), data aggregation (Xu et al. 2013) and broadcast (Xiao et al., 2014) in the SINR model.

When the link scheduling problem for connectivity is considered, the usual approach used by most algorithms divides the solution in two phases (Moscibroda, 2007), (Gong and Yang, 2013) and (Gogu et al., 2013). The first phase of heuristics constructs a connected graph topology. The second phase assigns power levels and time slots to each link of the graph such that the schedule length is minimized and all transmissions are received correctly, i.e., without violating the SINR at any receiver. In this paper we propose a different approach. The heuristic first assigns power levels such that a connected graph is built with minimum total transmission power. The second phase schedules links in each time slot, such that the time to schedule the connected graph is minimized and the SINR constraints are satisfied.

This paper is organized as follows. Section 2 introduces network model and problem formulation. In Section 3 we describe the algorithm used to build connected graphs with minimum total transmission power. Next, we present in Section 4 two heuristics to scheduling the connected graph given by the previous algorithm. Finally, in Section 5, we show and analyze some computational results. Section 6 concludes this study.

2. Network Model and Problem Formulation

Ad hoc sensor networks can be represented by a set V of transceivers (nodes) numbered from 0 to $|V|-1$, together with their locations. A transmission power p_u is assigned to each node $u \in V$. Each node can adjust its transmitting power, based on the distances to the receiving nodes and on the background noise. In the most common power attenuation model (Rappaport, 2001), the signal power falls with $1/d^\theta$, where d is the distance from the transmitter and θ is the path loss exponent.

For each ordered pair (u, v) of transceivers, with $u, v \in V$, there is a non-negative arc weight $d^\theta_{uv} q_v$, where $d_{uv} \geq 1, 0$ is the Euclidean distance between the transmitter u and the receiver v, and q_v is the receiver's power threshold for signal detection (usually normalized to 1). A signal transmitted by transceiver u can be received at node v if and only if $p_u \geq d^\theta_{uv}$. However, as nodes are communicating with each other in a common medium, the decodification of received messages can be affected by interference from concurrent transmissions.

Physical interference between links is minimized by constructing interference-aware routes. We adopt the physical interference model to characterize the interference. In the physical interference model, a received message sent from transmitter u is successfully decoded by v if the attenuated transmission power p_u/d^θ_{uv} exceeds the aggregate signal composed by the sum of the interference caused by all the other concurrently transmitting nodes. We assume there is only one available channel. According to this model, the transmission from node u is successfully decoded by node v if the signal-to-interference plus noise ratio (SINR) at v is equal to or larger than a given threshold β, that is

$$\frac{\dfrac{p_u}{d^\theta_{uv}}}{N_a + \displaystyle\sum_{w \in V\setminus\{u,v\}} \dfrac{p_w}{d^\theta_{wv}}} \geq \beta, \tag{1}$$

where N_a is the additive noise power.

Let $G(p) = (V, E(p))$ be an undirected communication graph, where $E(p) = \{[u, v] : u, v \in V, p_u \geq d^\theta_{uv}, p_v \geq d^\theta_{vu}\}$. A fundamental problem in wireless communications is to construct a connected communication graph $G(p)$ such that the constituent bidirectional links $[u, v] \in E(p)$ can be scheduled to fewest possible time slots assuming the SINR model of interference. A somehow dual problem to the scheduling problem is the power control problem (Katz et al. 2010) which consists in find minimum transmission powers p_u such that the link set $E(p)$ creates a connected communication graph.

We assume that the wireless channel is divided into t time-slots, $1 \leq t \leq \ell$, and that all nodes scheduled to transmit in the same time-slot do it simultaneously. A unidirectional link (u, v), such that $[u, v] \in E(p)$, is scheduled to time-slot s^t, $1 \leq t \leq \ell$, if the power level of node u in time-slot s^t is set to $p^t_u = p_u \geq d^\theta_{uv}$. A time-slot s^t is feasible if Condition (1) holds for each link $(u, v) \in s^t$, $[u, v] \in E(p)$. A schedule $S = \{s^1, s^2, \ldots, s^\ell\}$ is feasible if all $s^t \in S$ are feasible.

Formally, the power-efficient link scheduling problem for connectivity in wireless sensor networks (PSC-WSN) is, given the node set V, the distance d_{uv} for any $u, v \in V$, the path loss exponent θ, and parameters N_a and β, finding an assignment of transmission powers $p : V \to R+$ to every node $u \in V$ and $p \in \{0, p_{max}\}$, and a feasible schedule S of minimal length ℓ such that the resulting communication graph $G(p) = (V, E(p))$ is connected.

We describe in Algorithm 1 a heuristic to solve the PSC-WSN problem. Given the node set V and arc weights d^θ_{uv} for any $u, v \in V$, we first run in line 1 an algorithm to construct the minimum power connected communication graph $G(p) = (V, E(p))$, and then we create the link scheduling S for the link set $E(p)$ using a minimum-length link scheduling algorithm in line 2. The 2-approximation algorithm proposed by Cheng et al. (2003) presented in Section 3 is used to generate the minimum power connected communication graph $G(p) = (V, E(p))$. In Section 4, we describe two minimum-length link scheduling algorithms.

Algorithm 1 Power-Efficient Link Scheduling Heuristic – (PLSH)

Input: The node set V and arc weights d_{uv}^{θ} for any $u, v \in V$.
Output: A sequence $S = \{s^1, s^2, \ldots, s^{\ell}\}$ of time-slots with length ℓ.

```
Assign transmissions powers p such that graph G(p) = (V, E(p)) is connected;
Create the minimum-length link scheduling S using the link set E(p) as input;
return S;
```

---

3. Minimum Power Algorithm

The problem of joint transmission power assignment and link scheduling in order to build a connected communication graph in sensor networks may be solved by applying a topology control algorithm, and then identify the links that can transmit simultaneously assuring interference free transmissions under SINR constraints. Topology control is one of the most important techniques used in wireless ad hoc and sensor networks to reduce energy consumption. Algorithms for topology control attempt to reduce the number of links and the power consumption in a network subject to connectivity constraints (Moraes et al. 2009).

Given the node set V and arc weights d_{uv}^{θ} for any $u, v \in V$, the connected minimum power consumption problem consists in finding an assignment of transmission powers p_u to every node $u \in V$, such that the total power consumption $\sum_{u \in V} p_u$ is minimized and the resulting transmission graph is connected. The connected minimum power consumption problem was proposed in (Blough et al., 2002) and (Calinescu et al., 2002). Cheng et al. (2003) showed the importance of the problem in the case of sensor networks, proved its NP-completeness, and proposed two approximate algorithms.

In the following we describe our implementation of the 2-approximation algorithm proposed by (Cheng et al., 2003), named Incremental Power Greedy Heuristic, to generate solutions to the connected minimum power consumption problem. These solutions are used as input to the minimum-length link scheduling algorithms presented in Section 4.

Algorithm 2 Incremental Power Greedy Heuristic – (IPGH)

Input: The node set V and arc weights d_{uv}^{θ} for any $u, v \in V$.
Output: A sequence $S = \{s^1, s^2, \ldots, s^{\ell}\}$ of time-slots with length ℓ.

```
1: p ← 0, ∀u ∈ V;
2: Initialize H(p) = (V', E(p)) such that E(p) ← ∅ and V' ← {r} where r ∈ V;
3: while V' ≠ V do
4:     [u, v] ← ExtractMin(V, V') such that u ∈ V' and v ∈ V \ V';
5:     E(p) ← E(p) ∪ [u, v];
6:     V' ← V' ∪ {v};
7:     p_u, p_v ← UpdatePower([u, v]);
8: end while
9: return p;
```

Given the nodes set V located in the Euclidean plane, Algorithm 2 builds a connected graph one node at a time. Given the node set V and non-negative arc weights d_{uv}^{θ} for any $u, v \in V$, the algorithm sets $p_u = 0$ for all $u \in V$ in line 1. In line 2 it initializes a working graph $H(p) = (V', E(p))$ with $V' = \{r\}$ and $E(p) = \{[u, v] : u \in V', v \in V', p_u \geq d_{uv}^{\theta}, p_v \geq d_{vu}^{\theta}\} = \emptyset$, where $r \in V$ is any randomly selected initial node. Each iteration of the loop in lines 3 to 8 removes a node from set $V \setminus V'$ and inserts it to set V', until $V' = V$.

The procedure `ExtractMin()` in line 4 finds an edge $[u, v]$ with minimum greedy cost function $g(u, v)$.The greedy function that guides the construction is based on the wireless multicast property (Moraes and Ribeiro, 2013): if p_u is the current power assignment to node $u \in V'$

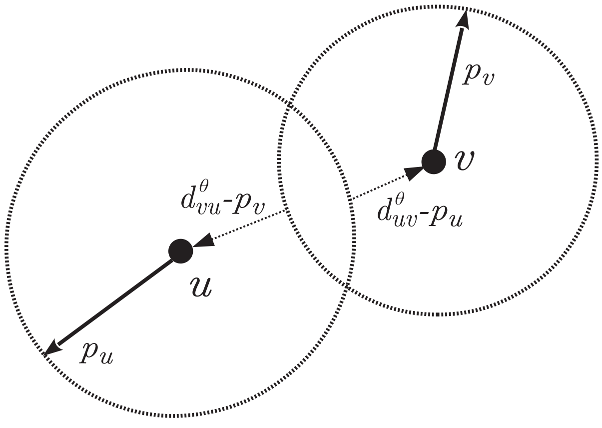

Figure 1: Example of how to compute the greedy cost function with $p_u > p_v$. Since $d^{\theta}_{uv} = d^{\theta}_{vu}$, we have $d^{\theta}_{uv} - p_u < d^{\theta}_{vu} - p_v$.

and there is a node $v \in V \setminus V'$ such that $d^{\theta}_{uv} > p_u$, then the incremental power required to set up communication from u to v is $d^{\theta}_{uv} - p_u$ (see Figure 1). Therefore, the greedy cost function is $g(u, v) = \max\{0, d^{\theta}_{uv} - p_u\} + \max\{0, d^{\theta}_{vu} - p_v\}$. If $g(u, v) = 0$, then the bidirectional communication between u and v is already set up.

Given the edge $[u, v]$, $u \in V$ and $v \in V \setminus V'$, with minimum greedy cost function $g(u, v)$ selected in line 4, the set of nodes V' and the set of edges $E(p)$ are updated, respectively, in lines 5 and 6. Procedure `UpdatePower()` in line 7 updates the power p_u needed for u to reach v and the power p_v needed to v reach u.We return, in line 9, the power assignment p, such that the communication graph $H(p) = (V', E(p)) = G(p)$ is connected.Algorithm 2 takes time $O(|V|^3)$ (Cheng et al., 2003).

4. Minimum-Length Link Scheduling Algorithms

Given a set of transmission link demands, the goal of an algorithm solving the minimum-length link scheduling problem is to schedule all the links into a minimum number of time-slots. This problem has been proved to be NP-complete in (Goussevskaia et al., 2007).

In the PSC-WSN problem, we consider as link demands the set of links $E(p)$ from the power efficient connected communication graph $G(p) = (V, E(p))$, built by Algorithm 2 as described in Section 3. The aim of the minimum-length link scheduling algorithm is to generate a sequence $S = \{s^1, s^2, \ldots, s^{\ell}\}$ of time-slots s^t, $1 \le t \le \ell$, with shortest length ℓ, such that all unidirectional links (u, v), such that $[u, v] \in E(p)$, are scheduled successfully at least once. The minimum-length link scheduling algorithm also assures that the SINR level is above a threshold β (see Equation 1) at every intended receiver in each time-slot. Before show the algorithms we give some definitions.

Definition 4.1 (Link Tolerance) *(Yang et al., 2010) The tolerance τ_{uv} of a unidirectional link (u, v) indicates how much interference a link can endure before the SINR falls below the threshold β. It is calculated by*

$$\tau_{uv} = \frac{\frac{p_u}{d^{\theta}_{uv}}}{\beta} - N_a$$

Definition 4.2 (Residual Link Tolerance) *(Yang et al., 2010) If the unidirectional link (u, v) was scheduled in time slot s^t, i.e. $p^t_u = p_u \ge d^{\theta}_{uv}$, then the residual tolerance $re\_\tau^t_{uv}$ of the unidirectional link (u, v) would indicate how much more interference the link can endure before the SINR* *falls below the threshold β. It can be calculated by*

$$re\_\tau^t_{uv} = \tau_{uv} - \sum_{(w,z)\in s^t \setminus \{(u,v)\}} \frac{p^t_w}{d^\theta_{wv}}$$

Definition 4.3 (Feasible Link) *(Yang et al., 2010) A unidirectional link (u, v) is* feasible *with respect to a time-slot s^t if, after the scheduling of (u, v) to the time-slot s^t, Equation 1 is satisfied for all $(w, z) \in s^t$.*

The greedy heuristic algorithm called Iterated Maximum Tolerance-to-Interference-Ratio (IMTIR) proposed by (Yang et al., 2010), and described in Algorithm 3, solves the minimum-length link scheduling problem filling the time slots one by one. The idea is schedule one link at a time. At each iteration of the algorithm, it looks for the link that is more tolerant and will cause less interference in the links scheduled so far to the current time slot. This property is achieved picking up the link (u, v) with maximum ratio $\frac{re\_\tau^t_{uv}}{max_{(w,z)\in s^t} \frac{p^t_u}{d^\theta_{uz}}}$. The combination of Algorithms 2 and 3 gives the solution for the PSC-WSN problem

Algorithm 3 receives, as input, the power assignment p computed by Algorithm 2 such that the communication graph $G(p) = (V, E(p))$ is connected. In line 1, the sequence of time-slots S is initialized as empty and the first current time-slot is created. In line 2 the unidirectional links (u, v) are extracted from the respective bidirectional links $[u, v] \in E(p)$ and stored in the unscheduled link set $A(p)$.Each iteration of the loop from line 3 to 18 fills the current time-slot s^i until no more unscheduled links in $A(p)$ could be scheduled to s^i, then a new one is created. This process is repeated until all links in $A(p)$ have been scheduled.

Algorithm 3 Iterated Maximum Tolerance-to-Interference-Ratio (MTIR)

Input: Transmissions powers p such that graph $G(p) = (V, E(p))$ is connected.
Output: A sequence $S = \{s^1, s^2, \ldots, s^\ell\}$ of time-slots with length ℓ.

1: $S \leftarrow \emptyset$; $i \leftarrow 1$;
2: $A(p) \leftarrow (u, v)$ and $A(p) \leftarrow (v, u), \forall [u, v] \in E(p)$;
3: **while** $A(p) \neq \emptyset$ **do**
4:     $A^i(p) \leftarrow A(p)$;
5:     $s^i \leftarrow \emptyset$; $p^i_u \leftarrow 0, \forall u \in V$;
6:     Compute τ_{uv} for each $(u, v) \in A^i(p)$, and set $re\_\tau^i_{uv} \leftarrow \tau_{uv}$;
7:     Pick $(u, v) \in A^i(p)$ with maximum τ_{uv}
8:     **repeat**
9:         $s^i \leftarrow s^i \cup \{(u, v)\}$; $\triangleright\ p^i_u = p_u \geq d^\theta_{uv}$
10:         $A^i(p) \leftarrow A^i(p) \setminus \{(u, v)\}$;
11:         $A(p) \leftarrow A(p) \setminus \{(u, v)\}$;
12:         Update $re\_\tau^i_{uv}$ for each link $(u, v) \in s^i$ and $(u, v) \in A^i(p)$;
13:         Remove every link $(u, v) \in A^i(p)$ that are not feasible with respect to s^i;
14:         Pick a feasible link $(u, v) \in A^i(p)$ with maximum $\frac{re\_\tau^i_{uv}}{max_{(w,z)\in s^i} \frac{p^i_u}{d^\theta_{uz}}}$;
15:     **until** $A^i(p) = \emptyset$
16:     $S \leftarrow S \cup \{s^i\}$;
17:     $i \leftarrow i + 1$;
18: **end while**
19: **return** S

In line 4, we initialize a working unscheduled link set $A^i(p)$, associated to the current time-slot s^i, as $A(p)$. The current time slot s^i, and the variables associated to it, are initialized in lines 5 and 6. We choose the link (u, v) with maximum link tolerance as the current link in line 7. The loop from line 8 to 15 fills the current time slot with the maximum number of links. It finishes when there is no links in the unscheduled link set $A^i(p)$. The current link (u, v) is scheduled to the current time-slot s^i in line 9 and removed from the both unscheduled link sets $A^i(p)$ and $A(p)$,respectively, in lines 10 and 11. The residual tolerance is updated in line 12 for all links (u, v) scheduled to the current time-slot, $(u, v) \in s^i$, and predicted for each unscheduled link $(u, v) \in A^i(p)$. The predicted residual tolerance is used to remove links that are not feasible with respect to s^i, i.e., every link $(u, v) \in A^i(p)$ with $re\_\tau^i_{uv} < 0$ is unfeasible with respect to s^i and removed from $A^i(p)$ in line 13.

We update the current link with the feasible link $(u, v) \in A^i(p)$ which has the maximum ratio $\frac{re\_\tau^i_{uv}}{max_{(w,z)\in s^i} \frac{p^i_u}{d^\theta_{uz}}}$ in line 14. We add the current time-slot s^i in the sequence S in line 16 and create a new time-slot in line 17. The final sequence $S = \{s^1, s^2, \ldots, s^\ell\}$ of time-slots with length ℓ is returned in line 19. (Yang et al., 2010) showed that the time complexity of Algorithm 3 is $O(m^3)$, where $m/2$ is the number of bidirectional links in $E(p)$. Since the connected communication graph $G(p) = (V, E(p))$ is a spanning tree, we have $|E(p)| = m/2 = O(n)$ and, consequently, $O(m^3) = O(n^3)$.

We have also implemented the algorithm named Maximum Bottleneck Tolerance (MBT) (Yang et al., 2010) to solve the minimum-length link scheduling problem. An unidirectional link (u, v) is a bottleneck of a time-slot s^t, $1 \le t \le \ell$, if it has the minimum residual tolerance, i.e., $re\_\tau^t_{uv} = min_{(w,z)\in s^t} re\_\tau^t_{wz}$. Let $T_b(s^t) = min_{(u,v)\in s^t} re\_\tau^t_{uv}$ denote the residual tolerance value of the time-slot s^t bottleneck.

The MBT heuristic is described in Algorithm 4. The combination of Algorithms 2 and 4 gives the solution for the PSC-WSN problem Yang et al. (2010) showed that, given $G(p) = (V, E(p))$ as input, the time complexity of Algorithm 4 is $O(n^4)$. This algorithm takes more time complexity than Algorithm 3, but shows better results. Each time Algorithm 4 tries to schedule a link, it should pick one such that the residual tolerance of the most vulnerable link is maximized after the selected link is scheduled to the current time-slot. By using this criteria, the MBT heuristic potentially increases the chance of scheduling more links in the current time-slot. The most vulnerable link is the time-slot bottleneck defined as the link with minimum residual tolerance value.

Algorithm 4 receives, as input, the power assignment p, such that the communication graph $G(p) = (V, E(p))$ is connected, computed by Algorithm 2.In line 1, the sequence of time-slots S is initialized as empty and the first current time-slot is created. In line 2 the unidirectional links (u, v) are extracted from the respective bidirectional links $[u, v] \in E(p)$ and stored in the unscheduled link ordered list $A(p)$. In line 4, the algorithm sorts the links in $A(p)$ in ascending order of their link tolerances. Each iteration of the loop from line 5 to 18 fills the current time-slot s^i until no more unscheduled links in $A(p)$ could be scheduled to s^i, then a new one is created. This process is repeated until all links in $A(p)$ have been scheduled.

In the loop from line 5 to 18, we first initialize the current time slot s^i as empty in line 6. Then, the first link (u, v) from the ordered unscheduled link list $A(p)$ is scheduled to the current time-slot s^i in line 7 and removed from $A(p)$ in line 8. After we add the first link in the current time-slot s^i, we calculate the set $A^i(p)$ of feasible links with respect to s^i in line 9. The loop from line 10 to 15 fills the current time slot with the maximum number of links. It is repeated until there is no unscheduled link left.

In the loop from line 10 to 15, we iteratively pick the link (u, v) from $A^i(p)$ that will give the maximum bottleneck tolerance if scheduled (line 11). Next, the selected link (u, v) is scheduled to s^i, removed from the unscheduled link set $A^i(p)$, and from the ordered unscheduled link list $A(p)$ in line 12.The residual tolerance for each unscheduled link in $A^i(p)$ is updated in line 13. If the residual tolerance of a link falls below 0, it becomes unfeasible with respect to the current time-slot s^i and we remove it from the unscheduled link set $A^i(p)$.

Algorithm 4 Maximum Bottleneck Tolerance (MBT)

Input: Transmission powers p such that graph $G(p) = (V, E(p))$ is connected.
Output: A sequence $S = \{s^1, s^2, \ldots, s^\ell\}$ of time-slots with length ℓ.

1: $S \leftarrow \emptyset$, $i \leftarrow 1$;
2: $A(p) \leftarrow (u, v)$ and $A(p) \leftarrow (v, u), \forall [u, v] \in E(p)$;
3: Compute τ_{uv} for each $(u, v) \in A(p)$;
4: Sort the links in $A(p)$ in ascending order based on their link tolerances τ;
5: **while** $A(p) \neq \emptyset$ **do**
6: $\quad s^i \leftarrow \emptyset$;
7: $\quad s^i \leftarrow s^i \cup$ {first unscheduled link $(u, v) \in A(p)$};
8: $\quad A(p) \leftarrow A(p) \setminus \{(u, v)\}$;
9: $\quad A^i(p) \leftarrow$ feasible links from $A(p)$ with respect to s^i;
10: $\quad$ **while** $A^i(p) \neq \emptyset$ **do**
11: $\quad\quad$ Pick link $(u, v) \in A^i(p)$ such that $re\_\tau_{uv}^t = max_{(w,z) \in A^i(p)} T_b(s^i \cup \{(w, z)\})$;
12: $\quad\quad s^i \leftarrow s^i \cup \{(u, v)\}$; $A^i(p) \leftarrow A^i(p) \setminus \{(u, v)\}$; $A(p) \leftarrow A(p) \setminus \{(u, v)\}$;
13: $\quad\quad$ Update the $re\_\tau_{uv}^t$ for each $(u, v) \in A^i(p)$;
14: $\quad\quad$ Remove every link $(u, v) \in A^i(p)$ that are not feasible with respect to s^i;
15: $\quad$ **end while**
16: $\quad S \leftarrow S \cup \{s^i\}$;
17: $\quad i \leftarrow i + 1$;
18: **end while**
19: **return** S ;

We add the current time-slot s^i in the sequence S in line 16 and create a new time-slot in line 17. The final sequence $S = \{s^1, s^2, \ldots, s^\ell\}$ of time-slots with length ℓ is returned in line 19.

5. Results

Computational experiments have been carried out on two classes of randomly generated test problems with 100 to 1600 nodes:

- Variable nodes density: the nodes are uniformly distributed in a square grid with fixed dimensions $D \times D$ with $D \in [10, 800]$.
- Constant nodes density: the square grip size is adjusted to keep density constant at 1 node per 10 square unit.

In both classes the Euclidean distance d_{uv} between nodes $u, v \in V$ is known, the path loss exponent θ is set at 4, the SINR threshold β is set at 16 and the additive noise power $N_a = 10^{-9}w$. For each problem size and type, 10 test instances have been generated. In all experiments, given one test instance, we run first the IPGH algorithm to compute a minimum power assignment such that the resulting communication graph is connected. In the following, the IMTIR and MBT heuristics are applied to schedule the set of links established by the connected communication graph given by IPGH. We evaluate the effectiveness of IMTIR and MBT heuristics in terms of their solution quality.

An Intel Core i5 machine with a 2.50 GHz clock and 6 Gbytes of RAM memory running Windows 7 Ultimate was used in all experiments. The heuristics were coded in C++ and compiled with the GNU 4.x Cygwin.

For the square grid size $D = \{10, 25, 50, 100, 200, 400, 800\}$ and problem dimension $|V| = 1000$, Table 1 shows the schedule length achieved by algorithms IMTIR and MBT, and the improvements in percent obtained by the MBT heuristic with respect to the solution values provided by IMTIR heuristic. They use as input the assignment of transmission powers p provided

Table 1: Average schedule length for algorithms IMTIR and MBT and problem dimension $|V| = 1000$.

	Schedule length		
Grid size	IMTR	MBT	Impr. (%)
10×10	44,0	37,0	15,91
25×25	44,2	37,1	16,06
50×50	44,6	37,1	16,82
100×100	44,3	37,8	14,67
200×200	44,3	36,7	17,16
400×400	44,4	37,6	15,32
800×800	44,0	37,1	15,68

by algorithm IPGH. We observe that the average schedule lengths computed by MBT are shorter than those computed by IMTIR independent of the network density. In particular, even with a large variation in density, MBT outperformed IMTIR within a small range reduction in schedule length, ranging from 14,67% to 17,16%. Figure 2 also shows that the variation in density does not affect the schedule length improvement obtained by MBT with respect to IMTIR.

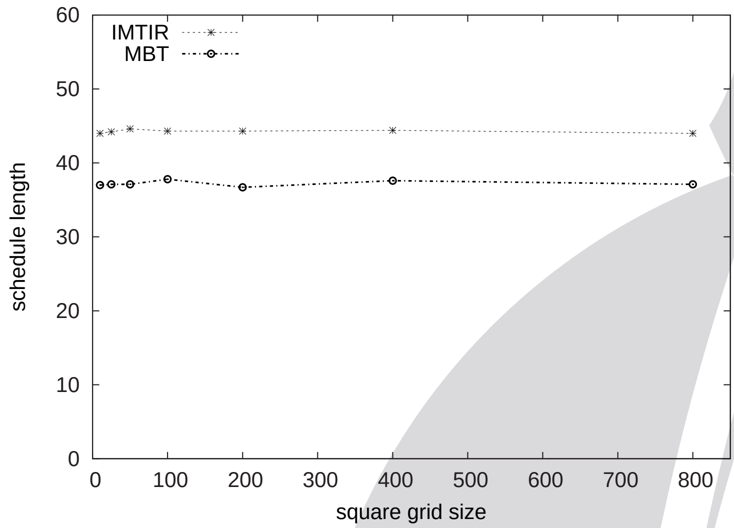

Figure 2: Schedule length comparison in function of the nodes density for problem dimension $|V| = 1000$.

In the next experiments, we focus our analysis into the constant nodes density case, since it does not affect the relative comparison between heuristics MBT and IMTIR. Given a problem dimension, the square grip size is adjusted to keep density constant at 1 node per 10 square unit. For instance, if $|V| = 1000$ then $D = 100$.

For each problem dimension $|V| = \{100, 200, 400, 800, 1600\}$, Table 2 displays the average schedule length given by algorithms IMTIR and MBT with different assignment of transmission powers. In this experiment, besides using the minimum assignment of transmission powers p provided by algorithm IPGH, IMTIR and MBT heuristics use, one at a time, the minimum power p increased by 25%, 50%, 75% and 100%.

Table 2 shows that, for both algorithms IMTIR and MTB, the schedule length built with the minimum assignment of transmission powers p can be improved when the transmission powers increases. These results show that the two objectives, power minimization and schedule length minimization, are contradictory. Considering the SINR model of interference, if the transmit power is high, the ongoing transmission may tolerate interference better because of a higher SINR. Instead of using the minimum power to maintain network connectivity, a high power level can substantially

Table 2: Average schedule length for algorithms IMTIR and MBT with power ranging from p to $p + 100\%$ and constant nodes density.

IMTIR									
	p	$p+25\%$		$p+50\%$		$p+75\%$		$p+100\%$	
$\|V\|$	SL	SL	Impr. (%)	SL	Impr. (%)	SL	Impr. (%)	SL	Impr. (%)
100	35,0	30,8	12,00	29,4	16,00	28,1	19,71	27,7	**20,86**
200	39,1	34,0	13,04	31,7	18,93	30,9	20,97	29,4	**24,81**
400	40,9	35,7	12,71	32,7	20,05	31,6	22,74	30,0	**26,65**
800	43,2	37,7	12,73	34,7	19,68	33,5	22,45	32,2	**25,46**
1600	46,5	38,3	17,63	35,7	23,23	34,6	25,59	33,0	**29,03**
MBT									
	p	$p+25\%$		$p+50\%$		$p+75\%$		$p+100\%$	
$\|V\|$	SL	SL	Impr. (%)	SL	Impr. (%)	SL	Impr. (%)	SL	Impr. (%)
100	27,3	24,4	10,62	23,7	13,19	22,5	17,58	22,1	**19,05**
200	32,2	28,3	12,11	25,9	19,57	25,6	20,50	25,1	**22,05**
400	34,2	29,6	13,45	27,8	18,71	26,8	21,64	26,0	**23,98**
800	36,2	32,3	10,77	29,8	17,68	28,7	20,72	28,5	**21,27**
1600	37,2	32,2	13,44	30,4	18,28	29,6	20,43	28,6	**23,12**

reduce the effect of concurrently transmitting nodes and thus improve the number of transmissions per time slot.

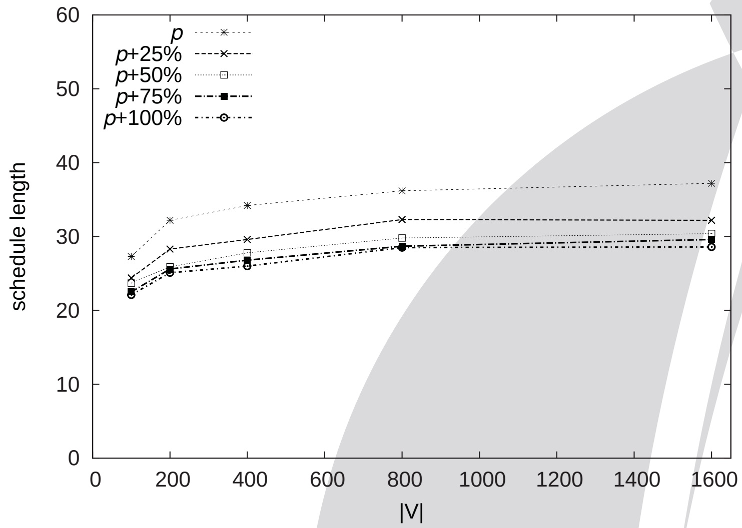

Figure 3: MBT schedule length solution values with increasing transmission powers.

Figure 3 also shows for the MBT heuristic that, as the transmission powers grows, the average schedule length always decreases, since more links can be scheduled in each time slot when links are less sensitive to interference.

In Table 3 we compare the average schedule length given by algorithms IMTIR and MBT under different assignment of transmission powers. The transmission powers are fixed, one at a time, at p, at p increased by 50%, and at p increased by 100%. The results presented in Table 3 show that the MBT heuristic results in better schedule length improvement with respect to IMTIR heuristic when using the minimum assignment of transmission powers p. As we can observe, the MBT improvement has the tendency to decrease for bigger values of transmission powers.

6. Conclusion

We considered the problem of joining the connected communication graph scheduling and power problems in the power-efficient link scheduling problem for connectivity in wireless sensor

Table 3: Average schedule length for algorithms IMTIR and MBT with transmission powers p, $p + 50\%$ and $p + 100\%$.

	p			$p+50\%$			$p+100\%$		
$\|V\|$	IMTR	MBT	Impr. (%)	IMTR	MBT	Impr. (%)	IMTR	MBT	Impr. (%)
100	35,0	27,3	**22,00**	29,4	23,7	19,39	27,7	22,1	20,22
200	39,1	32,2	17,65	31,7	25,9	**18,30**	29,4	25,1	14,63
400	40,9	34,2	**16,38**	32,7	27,8	14,98	30,0	26,0	13,33
800	43,2	36,2	**16,20**	34,7	29,8	14,12	32,2	28,5	11,49
1600	46,5	37,2	**20,00**	35,7	30,4	14,85	33,0	28,6	13,33

networks which consists in finding a set of power efficient links and scheduling then assuring interference free transmissions in order to achieve minimum schedule length and graph communication connectivity for a given placement of wireless nodes.

We then divided the problem into two subproblems and provided very simple and quick greedy algorithms to find good approximate solutions to real-life sized problems. Only variants concerned with first constructing a connected graph and then assigning lower power levels with minimum number of time slots has been tackled to date. Differently, we present for the first time in this context, a power efficient assignment during the graph topology construction and then the time slot scheduling with minimum length using the combination of two unrelated works (Cheng et al., 2003) and (Yang et al., 2010).

Different implementation strategies have been considered and compared in the quest for algorithm effectiveness. We have conducted extensive simulations and the results demonstrate that the link scheduling strategy used by MBT is more capable of reducing the schedule length for connected topologies under different node densities. The computational results also showed that the increasing in the transmission powers improves the length schedule for all algorithms.

Acknowledgments

This work was partially supported by CNPq (461286/2014-9 and 449369/2014-5).

References

Akyildiz, I., Sua, W., Sankarasubramaniam, Y., and Cayirci, E. (2002). Wireless sensor networks: a survey. *Computer Networks*, 38:393–422.

An, M. K., Lam, N. X., Huynh, D. T., and Nguyen, T. N. (2012). Connectivity in wireless sensor networks in the SINR model. In *Proceedings of the 20th Annual IEEE International Symposium on Modeling, Analysis and Simulation of Computer Systems*, pages 145–152, San Francisco.

Avin, C., Lotker, Z., Pasquale, F., and Pignolet, Y.-A. (2009). A note on uniform power connectivity in the SINR model. In *Algorithmic Aspects of Wireless Sensor Networks*, pages 116–127.

Benkert, M., Gudmundsson, J., Haverkort, H., and Wolff, A. (2008). Constructing minimum inter- ference networks. *Computational Geometry*, 40:17–194.

Blough, D. M., Canali, C., Resta, G., and Santi, P. (2009). On the impact of far-away interference on evaluations of wireless multihop networks. In *Proceedings of the 12th ACM international conference on Modeling, analysis and simulation of wireless systems*, pages 90–95, Tenerife.

Blough, D. M., Leoncini, M., Resta, G., and Santi, P. (2002). On the symmetric range assignment problem in wireless ad hoc networks. In *Proceedings of the IFIP 17th World Computer Congress - TC1 Stream*, pages 7–82, Montreal.

Borbash, S. and Ephremides, A. (2006). Wireless link scheduling with power control and SINR constraints. *IEEE Transactions on Information Theory*, 52:5106–5111.

Calinescu, G., Mandoiu, I. I., and Zelikovsky, A. (2002). Symmetric connectivity with minimum power consumption in radio networks. In *Proceedings of the IFIP 17th World Computer Con- gress - TC1 Stream*, pages 119–130, Montreal.

Cardoso, D. G. and Moraes, R. E. N. (2014). Minimização de retransmissores em redes de sensores sem fio com limite de alcance de rádio transmissão. In *Proceedings of the 32th Simpósio Brasileiro de Redes de Computadores e Sistemas Distribuídos*, pages 397–410, Florianópolis.

Cheng, X., Narahari, B., Simha, R., Cheng, M. X., and Liu, D. (2003). Strong minimum energy topology in wireless sensor networks: NP-Completeness and heuristics. *IEEE Transactions on Mobile Computing*, 2:248–256.

Gogu, A., Chatterjea, S., Nace, D., and Dilo, A. (2013). The problem of joint scheduling and power assignment in wireless sensor networks. In *Proceedings of the IEEE 27th International Conference on Advanced Information Networking and Applications*, pages 348–355, Barcelona.

Gong, D. and Yang, Y. (2013). Low-latency SINR-based data gathering in wireless sensor networks. In *Proceedings of the 32nd IEEE International Conference on Computer Communications*, pages 1941–1949, Turin.

Goussevskaia, O., Oswald, Y. A., and Wattenhofer, R. (2007). Complexity in geometric SINR. In *Pro- ceedings of the 8th ACM international symposium on Mobile ad hoc networking and computing*,pages 100–109, Montreal.

Halldórsson, M. M. and Mitra, P. (2012). Wireless connectivity and capacity. In *Proceedings of the Symposium on Discrete Algorithm*s, pages 516–526, Kyoto, Japan.

Katz, B., Volker, M., and Wagner, D. (2010). Energy efficient scheduling with power control for wireless networks. In *Proceedings of the 8th International Symposium on Modeling and Optimization in Mobile, Ad Hoc and Wireless Networks*, pages 160–169, Avignon.

Moraes, R. E. N. and Ribeiro, C. C. (2013). Power optimization in ad hoc wireless network topology control with biconnectivity requirements. *Computers & Operations Research*, 40:3188–3196.

Moraes, R. E. N., Ribeiro, C. C., and Duhamel, C. (2009). Optimal solutions for fault-tolerant topology control in wireless ad hoc networks. IEEE *Transactions on Wireless Communications*, 8:5970–5981.

Moscibroda, T. (2007). The worst-case capacity of wireless sensor networks. In *Proceedings of the 6th international conference on Information processing in sensor networks*, pages 1–10.

Moscibroda, T. and Wattenhofer, R. (2006). The complexity of connectivity in wireless networks. In *Proceedings of the 25th IEEE Conference on Computer Commun.*, pages 25–37, Barcelona.

Rappaport, T. (2001). *Wireless Communications: Principles and Practice. Prentice Hall*, Upper Saddle River, NJ, USA, 2nd edition.

Santi, P. (2005). *Topology control in wireless ad hoc and sensor networks*. John Wiley & Sons, Ltd., West Sussex.

Xiao, S., Pei, J., Chen, X., and Wang, W. (2014). Minimum latency broadcast in the SINR model: A parallel routing and scheduling approach. *IEEE Communications Letters*, 18:1027–1030.

Xu, X., Li, X.-Y., and Song, M. (2013). Efficient aggregation scheduling in multihop wireless sensor networks with SINR constraints. *IEEE Transactions on Mobile Computing*, 12:2518–2528.

Yang, D., Fang, X., Xue, G., Irani, A., and Misra, S. (2010). Simple and effective scheduling in wireless networks under the physical interference model. In *Proceedings of the IEEE Global Telecommunications Conference*, pages 1–5, Miami.