Joint Routing and Scheduling via Iterative Link Pruning in Wireless Mesh Networks

Vasilis Friderikos[1]
Katerina Papadaki[2]

[1]Centre for Telecommunications Research, King's College London
[2]Operational Research Group, Department of Management, London School of Economics

Working Paper LSEOR 07.94 ISBN: 978 0 85328 049 1

First published in Great Britain in 2007
by the Operational Research Group, Department of Management
London School of Economics and Political Science

Typeset, printed and bound by:
The London School of Economics and Political Science
Houghton Street
London WC2A 2AE

Working Paper LSEOR 07.94 ISBN: 978 0 85328 049 1

Joint Routing and Scheduling via Iterative Link Pruning in Wireless Mesh Networks

Vasilis Friderikos, Katerina Papadaki

Abstract

The focus of this paper is on routing in Wireless Mesh Networks (WMNs) that results in STDMA schedules with minimum frame length. In particular we focus on spanning tree construction and formulate the joint routing, power control and scheduling problem as a Mixed Integer Linear Program (MILP). Since this is an $\mathcal{NP}$-complete problem, we propose an iterative pruning based routing scheme that utilizes scheduling information. Numerical investigations reveal that the iterative pruning algorithm outperforms previously proposed routing schemes that aim to minimize the transmitted power or interference produced without explicitly taking into account scheduling decisions.

Index Terms

Scheduling, Spatial Time Division Multiple Access, Routing

I. Introduction

Algorithmic aspects of wireless mesh networks (WMNs) are currently a vigorous area of research and have steadily accumulated momentum over the last few years. The leading exponents of this increased interest are the potential multifarious applications of WMNs [1]. Admittedly, the two most important of them being, low cost and rapid deployable broadband last mile connectivity to the Internet and provision of backhaul support for 3G cells and IEEE 802.11'x' hot spots.

Efficient resource utilization in WMNs calls for scheduling and routing policies that maximize the aggregate throughput of the system. Under this perspective, the central theme of this paper is the design of joint scheduling and shortest path spanning tree schemes that provide increased system performance. With a preconstructed spanning tree within the mesh network the cornerstone aim of the scheduling engine is either to maximize the transmission opportunities of active links in a specific time window (frame) by taking into account the interference caused by simultaneously transmitting nodes or to minimize the time span for all links to transmit. Concurrent transmissions is of utmost importance since they increase system efficiency but can lead to erroneously reception at the receiver, if the level of the received signal is too weak compared to the aggregate interference. Thus, the spatial reuse of timeslots heavily depends on the selected active set of links in the mesh topology. But, the active set of links is constructed by the routing algorithm. Therefore, and as it will become vividly clear in the sequel, there is an interplay between scheduling and routing decisions. The rationale of designing joint routing and scheduling schemes stems exactly from this interplay between the two functionalities.

The medium access control scheme considered hereafter is based on time division multiple access (TDMA), where time is divided into timeslots and each node can transmit only at predefined timeslots, thus collisions can be avoided [1]. Since nodes are spatially distributed, timeslots can be potentially reused by nodes that are sufficiently far apart. Spatial reusing of timeslots has been defined in the seminal work of Kleinrock [2] and is called Spatial-TDMA (STDMA).

A shorter version of this paper has been submitted to GLOBECOM 2007 Symposium on Wireless Networking

V.Friderikos is with the Centre for Telecommunications Research at King's College London, England
E-mail: vasilis.friderikos@kcl.ac.uk

K. Papadaki is with the Operational Research Group, Department of Management, London School of Economics
E-mail: k.p.papadaki@lse.ac.uk

[1] The same analysis can also be applied for FDMA based networks

In this paper we focus on utilizing shortest path algorithms, which are widely studied and used in practise. The emphasis is on Dijkstra's algorithm, which for bounded degree graphs finds the shortest paths from a source node to every other node in $\mathcal{O}(n \log n)$ time. The cost metric used is the required transmission power for a link (i, j) to be established. Then iterations are performed as follows: For each shortest path rooted spanning tree we eliminate the link which has the highest interference metric and subsequently calculate the minimum number of timeslots required using a greedy scheduling heuristic. At each iteration we check if the number of timeslots has been reduced, and the iterations are repeated based on a number of different stopping criteria.

As will be shown in later sections, the proposed iterative pruning algorithm outperforms previously proposed schemes where these two costs, required power for link establishment and interference produced, are either used individually or linearly combined to create a single metric.

A. Organization of the paper

The rest of the paper is organized as follows. In section II selected closely related previous works in the area of joint routing and scheduling are outlined. The problem description and the mixed integer linear program formulation are detailed in section III. The inherent interplay between routing and scheduling is explained in section IV. In section V suboptimal joint scheduling and routing schemes are explained and section VI outlines the proposed pruning algorithm. Numerical investigations are reported in section VII and finally, section VIII concludes the paper by outlining the main findings followed by a brief discussion on some interesting future avenues for research.

II. Review of Selected Prior Works

After the introduction of the Spatial-TDMA concept by Kleinrock in [2], general timeslots (or channels) assignment scheduling problems have been extensively studied in the literature. The bulk of previous research work focused on graph theoretic solutions by conceiving link scheduling as a graph coloring problem [3], [4], [5]. In the basic setting, graph coloring approaches aim to tackle the *primary* and *secondary* conflicts between links. More specifically, any pair of directed edges (a, b), (c, d) may be colored with the same color if and only if (i) a, b, c, d are all mutually distinct and (ii) edges (a, d), (c, b) do not belong in the set of edges in the graph. When the first (second) condition fails to hold, then there will be a primary (secondary) conflict between edges (a, b) and (c, d). Scheduling based on graph theoretic tools proved essential for formally defining the problem and for the design of distributed solutions. The limitations on the other hand of these solutions stem from the fact that the aggregate effect of interference of links transmitting in concurrent timeslots (reflected in the SINR ratio), is not taken explicitly into account [6]. In other words, a schedule provided by a graph coloring technique may lead to a non-feasible allocation when the SINR thresholds are taken into account. Related to this last point is the observation that an optimal schedule based on graph coloring can be considered as a lower bound on the minimum number of timeslots that can be used in the network.

To fill this void, the authors in [9] have explicitly taken into account the SINR thresholds together with power control for constructing minimum frame length scheduling in STDMA networks with directional antennas. From a computational complexity perspective, even without taking into account the aggregate interference, constructing a transmission schedule of timeslots where all links are scheduled with the minimum number of timeslots, i.e., minimum frame length, has shown to be an $\mathcal{NP}$-complete problem [7].

The work of Tasiulas et al., [8] showed that the capacity region of wireless multi hop networks depends on the power allocation vector (which itself depends on channel conditions), the routing and scheduling decisions. This formal characterization of the inherent coupling between power control, scheduling and routing, sparked a research interest in schemes that attempt to optimize them jointly [14], [15]. These, so called *cross-layer optimization* approaches have recently been extended to take into account end-to-end flow and congestion control decisions (transport layer) [16]. Polynomial complexity algorithms together with necessary and sufficient conditions for scheduling and routing of a pre-defined set of source-destination rates in mesh networks have been discussed in [17]. In contrast to these previous works the emphasis in this paper, is on how to construct spanning trees that minimize the frame length (in terms of required timeslots) in the mesh network.

Finally, it's worth mentioning that pruning techniques have been mainly used within QoS routing to produce a sparser graph, consisting entirely of feasible links [18], [19]. In other words, links are deleted from the topology if their available resources do not meet the corresponding constraints. In our case the incentive for link pruning is rather different; pruning is used to delete links that produce high interference to neighbor nodes that limits the possible spatial reuse of timeslots.

A. Contribution of the paper

To the authors best knowledge, this is the first paper that *explicitly* addresses the issue of how to jointly construct a spanning tree while minimizing the required frame length (in terms of timeslot) in a wireless mesh network. In that respect, the contributions of the paper represent measurable progress on the following fronts,

1) Formulation of the mixed-integer linear program to perform optimal jointly spanning tree construction and scheduling that minimizes the required frame length in timeslots.
2) Interference aware iterative pruning routing algorithm to construct spanning trees in the WMN with a minimum frame length schedule.
3) Quantification of the gains in terms of scheduling of the pruning scheme compared to previous proposed schemes based on an extensive set of simulations.

It is worth mentioning that even though in this paper we have assumed omni-directional antennas (0dB gain) and baseline path loss models, the proposed scheme is independent of the operational characteristics and models used. Thus, results drawn in this paper can be applied for different antenna radiation patterns and/or link gain models.

III. Problem Description

Before embarking our study of sub-optimal solutions in later sections, we first formulate the problem of joint routing and scheduling as a mixed linear integer program (MILP). In this paper, we assume that the sole purpose of the routing is to improve scheduling. Section III-A details the problem formulation for performing STDMA scheduling under the assumption of a pre-defined route and section III-B augments the scheduling model to incorporate routing decisions.

For performing joint scheduling and routing in wireless mesh networks we consider the graph G, defined by the (V, L) pair, where V is a set of vertices (wireless nodes) and L is the set of edges (transmission links that satisfy the SINR threshold criterion), i.e.

$$L = \{(u, v) | u, v \in V \text{ s.t. u can transmit to v and vice versa}\}. \quad (1)$$

Routing is usually performed using a weighting function $w : L \to \mathbb{R}$, which assigns a weight to each edge. The weight of an edge is commonly related with the required transmission power, which depends on the Euclidean distance between the nodes and the level of interference. A number of different possible edge weights that implicitly take into account scheduling information for sub-optimal routing and scheduling will be discussed in the next sections.

A. A Mixed Integer Linear Programming (MILP) Formulation for Scheduling

We first focus our attention on how to perform optimal scheduling decisions, under the assumption that routing paths are pre-constructed. In this case, the routing will create the directed graph $G_S = (V, L_S)$, where $L_S \subseteq L$, and scheduling will be performed on G_S. We further denote by $V_S^T \subseteq V$ and $V_S^R \subseteq V$ the set of transmitting and receiving nodes respectively.

We encapsulate power control within the MILP formulation by introducing the variable p_{ijt}, which express the transmitted power by node i in link (i,j) at timeslot t, under the constraint that $0 \leq p_{ijt} \leq P_{max},\ \forall\ t\ \in\ [1,M]$. The variable P_{max} express the power ceiling at the transmitting node (without loss of generality P_{max} is assumed to be equal for all nodes in the WMN). Additionally, we assume that omnidirectional antennas are used by all wireless nodes to transmit and receive signals. Thus, the interference level produced by link (i,j) to all other receiving nodes will be based on their Euclidean distance with node i. With a constant target bit error rate, i.e. $E_b/N_0 = \Gamma$, the transmission can be translated into a signal to interference ratio requirement, which will be denoted hereafter as γ. By W we denote the lump sum thermal noise power and by g_{ij} the link gain between nodes i, j, which encapsulates both path loss and slow fading.

To be able to express now the problem in a mathematical programming setting we introduce the boolean variables x_{ijt} and π_t, which are defined as follows,

$$x_{ijt} = \begin{cases} 1 & \text{if link } (i,j) \text{ active at timeslot t} \\ 0 & \text{otherwise} \end{cases} \tag{2}$$

$$\pi_t = \begin{cases} 1 & \text{if timeslot } t \text{ is used} \\ 0 & \text{otherwise} \end{cases} \tag{3}$$

The mixed-integer linear program for scheduling that minimize the required frame length in a pre-defined route on the set of links L_S is denoted as $OS(L_S)$ and can be written as follows,

$$\min \sum_{t=1}^{M} \pi_t$$

$$\text{subject to} \quad \pi_t \leq \sum_{(i,j)\in L_S} x_{ijt}, \quad \forall\ t \tag{4}$$

$$\sum_{(i,j)\in L_S} x_{ijt} \leq \pi_t \cdot \mid L_S \mid, \quad \forall\ t \tag{5}$$

$$\sum_{t=1}^{M} x_{ijt} \geq 1, \quad \forall (i,j) \in L_S \tag{6}$$

$$\sum_{j\,\in\, V_S^R} x_{ijt} + \sum_{k\,\in\, V_S^T} x_{kit} \leq 1\ \ \forall\, i \in V_S^T \cap V_S^R,\ \ \forall\, t \tag{7}$$

$$\frac{g_{ij} p_{ijt} + (1 - x_{ijt})\,\Lambda}{\sum\limits_{(m,n)\in L_S \backslash \{(i,j)\}} g_{mj} p_{mnt} + W} \geq \gamma,\ \ \forall\,(i,j) \in L_S,\ \ \forall\ t \tag{8}$$

$$x_{ijt} \leq \frac{p_{ijt} g_{ij}}{W\gamma}\ \ \forall\,(i,j) \in L_S,\ \ \forall\, t \tag{9}$$

$$x_{ijt} \geq p_{ijt}/P_{max}\ \ \forall\,(i,j) \in L_S,\ \ \forall\, t \tag{10}$$

$$x_{ijt} \in\ \{0,1\}\ \ \forall\,(i,j) \in L_S,\ \ \forall\ t \tag{11}$$

$$\pi_t \in\ \{0,1\}\ \ \forall\ t \tag{12}$$

$$0 \leq p_{ijt} \leq P_{max}\ \ \forall\,(i,j) \in L_S,\ \ \forall\ t \tag{13}$$

In this formulation an initial frame length M is assumed, where all links can be easily scheduled. For example an initial frame length value M could be the number of links.

Constraints (4) and (5) are the binding constraints for variables π_t and x_{ijt}. The requirement that all links transmit at least once during the frame length is ensured by constraint (6). Constraint (7) is the degree constraint, i.e., a node cannot transmit and receive at the same timeslot. Constraint (8) express the required SINR threshold that should be satisfied in order to have a successful reception at the receiver. The term $\Lambda(1 - x_{ijt})$ ensures that the inequality is satisfied when link (i, j) does not transmit at timeslot t, for a sufficiently high value of Λ. The binding constraints for variables x_{ijt} and p_{ijt} are shown in (9) and (10). These binding constraints ensure that if link (i, j) is not transmitting at timeslot t then the transmitted power p_{ijt} is zero and vice versa. Constraint (9) is based on the assumption that all links (i, j) in L_S satisfy the SINR constraint when there are no concurrent transmissions, which is equivalent to: $g_{ij}p_{ijt} > \gamma W$.

B. Performing Joint Scheduling and Routing

In the previous section we formulated the scheduling problem given a fixed routing L_S. Allowing flexibility with routing decisions can improve the resulting scheduling. The aim here is to construct a routing such that the number of timeslots in a time frame is minimized. We focus our routing decisions on constructing spanning trees, which are defined below:

Definition A tree is a subgraph of G that does not contain any cycles.

Definition A spanning tree of a graph G is a tree that contains all vertices of G.

We augment the previously defined scheduling model to incorporate both routing (tree construction) and scheduling decisions. Note that the optimal joint routing and scheduling problem operate on the graph $G = (V, L)$. Before describing the new constraints that need to be added, we first introduce the routing variables $y_{i,j}$, which are defined as follows:

$$y_{ij} = \begin{cases} 1 & \text{link } (i, j) \text{ in optimal spanning tree} \\ 0 & \text{otherwise} \end{cases} \tag{14}$$

Without loss of generality we assume that node r is the root node in the constructed spanning tree. Based on the above definitions, the optimal joint scheduling and spanning tree construction problem will be denoted as $OSR(L)$, which is based on the set of all feasible links L. The mathematical formulation of the $OSR(L)$ can be constructed by adding the following routing constraints to the already defined $OS(L)$ formulation.

$$y_{ij} \leq \sum_{t=1}^{M} x_{ijt} \leq y_{ij} \cdot M, \ \forall (i, j) \in L \tag{15}$$

$$\sum_{i \in V, i \neq j, (i,j) \in L} y_{ij} = 1 \ \forall j \neq r \tag{16}$$

$$\sum_{i \in V, i \neq r} y_{ir} = 0 \tag{17}$$

$$\sum_{(i,j) \in L} y_{ij} = \mid V \mid - 1 \ \forall (i, j) \in L \tag{18}$$

$$y_{ij} + y_{ji} \leq 1, \ \forall (i, j) \in L \tag{19}$$

Constraint (15) binds the boolean variables x_{ijt} and y_{ij} so that a link (i, j) transmits if and only if it belongs to the optimal spanning tree. Constraints (16), (17), (18) and (19) construct a directed spanning tree from node r to all other nodes in the network. A formal proof regarding the loop prevention constraints is detailed in [22].

The $OSR(L)$ formulation constructs a tree that produces schedules with the minimal timeslot frame length. Given that $OR(L_S)$ is an $\mathcal{NP}$-complete problem [7], the $\mathcal{NP}$-completeness of $OSR(L)$ follows as a corollary.

Corollary 3.1: The joint scheduling and spanning tree construction problem, $OSR(L)$, is $\mathcal{NP}$-complete.

IV. The Binding Nature of Spanning Tree Construction and Scheduling

The aim of this section is twofold. Firstly to reveal the closely coupled nature of the spanning tree construction and the scheduling problem. Secondly, this discussion will motivate the proposed scheme for spanning tree construction.

Figure 1 shows the worst case scenario of a minimum power spanning tree in terms of utilization of the timeslots. As shown in the figure, the transmission areas of the nodes are nested in the sense that each node's transmission area includes all nodes that are further away from the root node. If we define the transmission area of node i as A_i then this can be written as $\{i, i+1, i+2 \ldots\} \subseteq A_i$. This means that each node i can not transmit at the same timeslot as nodes $i+1, i+2, \ldots$. Thus, no concurrent transmission can occur and the number of timeslots required for all nodes to transmit grows linearly, $\Omega(n)$ with the number of transmitting nodes. On the other hand, figure 2 depict a topology where the minimum power spanning tree requires only two timeslots for all the nodes to transmit. Two timeslots is the minimum number required since the degree of the topology is two. As shown in the figure, two timeslots are sufficient since the transmission areas of nodes transmitting at timeslot one (or two) do not overlap. Note that this one dimensional topology has the minimum interference between nodes that transmit concurrently at timeslots one or two and in this case a new timeslot will only be required if the aggregate interference produced by the nodes transmitting at timeslot one or two produce a violation on the SINR threshold.

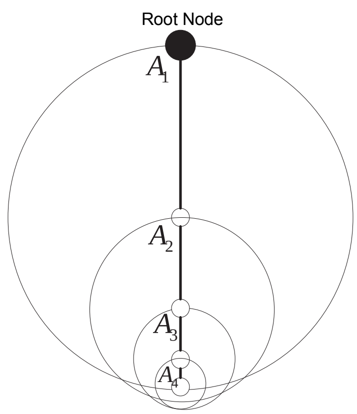


Fig. 1. Worst case scenario for timeslot reuse: The number of required timeslots is equal to the number of edges, i.e., $M = |L|$.

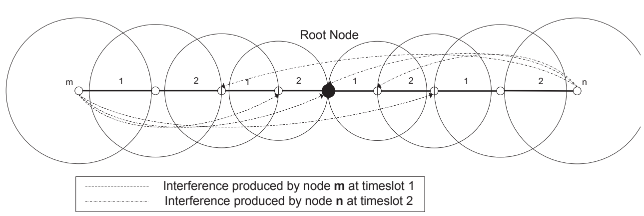


Fig. 2. Best case scenario for timeslot reuse: The number of required timeslots is equal to the number of edges, i.e., $M = |L|$.

In reality, we expect that wireless mesh network topologies will lie somewhere in between the worst and best case scenarios described above, therefore efficient algorithms that can provide high spatial reuse of timeslots become crucially important.

V. Shortest Path Tree Construction Schemes in WMN's

The joint routing and scheduling problem defined in section V is $\mathcal{NP}$-complete and thus intractable for realistic network sizes. Thus we turn our attention to existing routing algorithms and try to incorporate scheduling information into routing decisions.

In most widely used routing protocols for constructing trees, the paths are computed based on Dijkstra's algorithm to find shortest path spanning trees. The weight assigned to each link (i, j), $w(i, j)$, is usually taken to be proportional to the power needed to transmit on link (i, j). In the sequel we propose Dijkstra-based routing schemes that use different weights, with the aim of improving link scheduling. In section VII we evaluate the performance of these schemes and compare them to the proposed Interference Aware Pruning Routing scheme, described in section VI.

A. Minimum Power Routing - MPR

This scheme constructs shortest path spanning trees in $G = (V, L)$ from the root node r to all other nodes $V \setminus \{r\}$ using Dijkstra with transmitted power as a link cost. This cost results in reduction of the overall interference. Given that the transmitted power for link (i, j) relates to the distance between nodes i and j, $d(i, j)$, we define the following cost for MPR:

$$w_P(i, j) = d(i, j)^{\alpha}, \tag{20}$$

where α is the path loss exponent which varies between $2 - 4$.

In order to examine the effect of Dijkstra-based routing schemes on scheduling decisions, we assume in this section the following simple interference model: The interference caused during the transmission of link (i, j) only results in unsuccessful reception of nodes that lie within the disc with center i and radius $d(i, j)$. Any receiving nodes that lie outside the disc are unaffected. We call this model as disc-based interference.

Proposition 5.1: Assuming a disc-based interference model, the MPR scheme does not result in a schedule with minimum timeslot frame length.

Proof: We show this using a counter example. Figure 3(a) shows the shortest path spanning tree constructed by MPR for the given topology. In the MPR tree the transmission of link (i, m) is affecting five links (including the link that has a degree constraint with link (i, m)). These five links require four timeslots (minimum), and since none of them can be reused, the required number of timeslots should be increased by one to accommodate link (i, m). In the tree shown in 3(b) node m is connected via node j. In this case, the link (m, j) is affecting two links (the link from root node to node n, and link (n, j)), and therefore one of the four timeslots from the other branch of the tree can be reused.

The tree depicted in (b) is not a shortest path spanning tree with respect to w_P since the path from node m to the root node is longer than the equivalent path in (a). However, the tree in (b) produces a schedule with shorter frame length (in terms of timeslots). ∎

Shortest path spanning trees can be computed in polynomial time using the Dijkstra or Bellman-Ford algorithms. A brute force approach to find the tree with the minimum frame length would be to enumerate all possible trees and for each one perform optimal scheduling. Even without taking into account the embedded scheduling problem, enumerating all trees has exponential computational complexity due to proposition 5.2.

Proposition 5.2: (Cayleys Formula): The number of labeled trees on n vertices is n^{n-2}.

B. Minimum nearest Neighborhoods Routing - MNR

The MNR algorithm tries to minimize the number of nodes that are within the area of each link in the shortest path spanning tree. In order to compute such a tree Dijkstra's algorithm can be deployed where the cost of each link $(i, j) \in L$ is equal to the number of receiving nodes that are within its transmission range (taken to be the disc of center i and radius $d(i, j)$). In this case the cost can be written as follows,

$$w_N(i, j) = \sum_{n \in V \setminus \{i, j, r\}} \mathcal{I}_{(i,j)}(n) \tag{21}$$

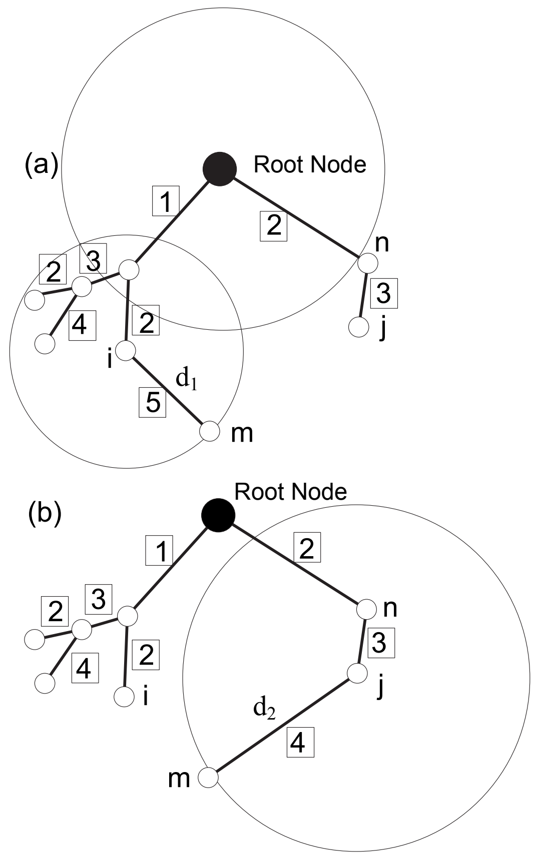


Fig. 3. (a) Minimum Power Spanning Tree (MPST),and (b) a spanning tree that requires less number of timeslots (better spatial reuse). Timeslots are shown within the rectangular boxes.

where $\mathcal{I}_{(i,j)}(n)$ is the indicator function which is defined as follows for $n \neq i, j$:

$$\mathcal{I}_{(i,j)}(n) = \begin{cases} 1 & \text{if } d(i,n) \leq d(i,j) \\ 0 & \text{otherwise} \end{cases} \tag{22}$$

This algorithm can also include a lower bound on the number of nodes that are within the transmission range of each link so that connectivity can be established with high probability [20].

A drawback of this scheme is that it may introduce edge-crossings in the constructed tree.

Proposition 5.3: The MNR scheme may create shortest path spanning trees that are non-planar graphs.

Proof: Figure 4 shows a possible construction of a spanning tree based on the MNR algorithm. The root node is node a and after the construction of links (a, b), which has cost 0, and (a, c), which has cost 1, the cost of the link (b, d) is 4 (nodes within the circle shown by solid lines) whilst the cost for links (a, d) and (c, d) is 5 and 7 respectively (nodes within the dashed and doted dashed circles respectively). Thus, the least cost path to node d is through node b and therefore an edge-crossing will be introduced. ∎

Link crossing can be detected and subsequently planarity can be restored but the current proposed techniques need to adapted before applied for tree construction (see [21] and references therein).

C. Interference based Routing - IR

In this case the actual interference that will be produced to the other receiving nodes in the network is taken into account to produce the cost for every link in the network. More specifically, the cost for link

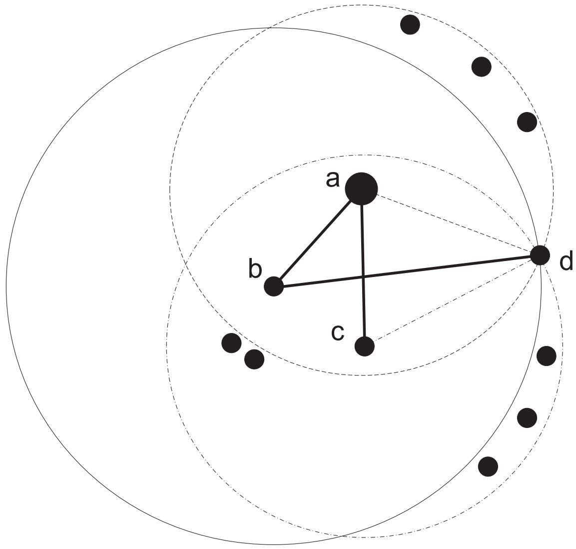


Fig. 4. Possible edge-crossing in the minimum nearest neighborhoods spanning tree algorithm

(i, j) is computed as follows,

$$w_I(i,j) = \frac{\sum_{n \in V \setminus \{i,j,r\}} g(i,n)}{g(i,j)} \tag{23}$$

Therefore, the cost for link (i, j) is inversely proportional to the link gain $g(i, j)$ but weighted with the aggregate link gains of node i to all other receiving nodes in the network. Thus, the actual interference that will be produced by link (i, j) is explicitly taken into account.

D. Weighted Power and Interference Routing - WPIR

In WPIR the two different metrics, i.e. required power for establishing the link and interference caused by the link, are condensed into a single metric via a linear combination. The cost for link (i, j) can be therefore written as follows,

$$w_{PI}(i,j) = \beta w_P(i,j)\Theta + (1-\beta)\, w_I(i,j) \tag{24}$$

where β controls the weight of each individual metric in the cost and Θ is a normalizing constant between the average w_P and w_I values. By this linear combination a single weight is assigned to every link and thus it becomes possible to use a Dijkstra-like algorithm. Since different spanning trees will be constructed with different values of β, a drawback of this scheme is that by linearly combining the two metrics, the optimal weighting value will be different for different topologies. This routing scheme is similar to the one discussed in [10].

VI. Interference aware Pruning Routing Algorithm - IAPR

The algorithm presented herein is based on an iterative version of the Dijkstra's shortest path algorithm. In each iteration of the algorithm links that produce the highest interference are pruned in later iterations of the algorithm. The idea is that by excluding links that produce severe interference the spatial timeslot reuse could be enhanced.

At each iteration k a shortest path spanning tree T_k is constructed, using weights w_P, based on the set of available links and scheduling is performed on T_k to find the minimum frame length S_k to schedule all links in the tree. The function $I_k(e)$ is a metric of interference produced by link e at iteration k in shortest path spanning tree T_k. The spanning tree is updated at each iteration by removing the link with the highest interference and running Dijkstra on the remaining links. We keep the spanning that produced a schedule with the minimum frame length. This continuous until the stopping criteria of the algorithm are satisfied. The pseudo code of the proposed IAPR scheme is shown in algorithm 1.

Algorithm 1 Interference Aware Pruning Routing - IAPR

$\overline{G}(V,\overline{L}) \leftarrow G(V,L)$, pre-processing (see section VI-B)
$k = 0$,
K, maximum number of iterations
T, best spanning tree found so far
S, minimum frame length achieved so far
$T_k \leftarrow \emptyset$, spanning tree at iteration k
$S_k = |\overline{L}|$, frame length achieved at iteration k by tree T_k
repeat
  if $k = 0$ **then**
    $T_{k+1} \leftarrow$ Dijkstra using w_P on $\overline{G}(V,\overline{L})$
  else
    $T_{k+1} \leftarrow$ Modified Dijkstra using w_P on $\overline{G}(V,\overline{L})$
  end if
  $S_{k+1} \leftarrow$ Schedule(T_{k+1})
  if $S_{k+1} < S_k$ **then**
    $T \leftarrow T_{k+1}$
    $S \leftarrow S_{k+1}$
  end if
  Find link $e \in \overline{L}$ such that $I_{k+1}(e) = max_{l\in\overline{L}}\{I_{k+1}(l)\}$
  $\overline{L} \leftarrow \overline{L} \setminus \{e\}$
  $k = k + 1$
  Calculate ε from equation (27)
until $(\varepsilon \leq \varepsilon_T) \vee (k > K)$
return T, S

A. Properties of the IAPR scheme

Since the pruning algorithm eliminates in each iteration links that have been previously used in constructing shortest path spanning trees, the aggregate shortest path cost will not decrease with iterations. This characteristic of the IAPR scheme is encapsulated in the following result. Let us denote by $p_k(i)$ the aggregate power for the shortest path in T_k from the root node to node i .

Proposition 6.1: If by $P_k = \sum_{i\in V\setminus\{r\}} p_k(i)$ we denote the aggregate transmitted power in tree T_k constructed by IAPR algorithm at iterations k, then

$$P_1 \leq P_2 \leq \ldots \leq P_K \tag{25}$$

Lemma 6.2: If link (i,j) is eliminated at iteration k of the interference aware pruning algorithm then, all nodes of tree T_k, where j is not a parent, will have the same predecessor in tree T_{k+1}.

Proof Since link $(i,j) \in T_k$, (i,j) belongs to the shortest path from root node to node j in T_k. Eliminating (i,j) at iteration k, the shortest path cost to node j in T_{k+1} will increase (or remain the same). Thus, $p_{k+1}(j) \geq p_k(j)$. Thus, any node that did not have node j as parent in tree T_k, will not have j as a parent in tree T_{k+1}. ∎

Lemma 6.2 indicates that trees T_k and T_{k+1} may have a large set of common links. This observation motivates the following modification of the Dijkstra algorithm:

Definition The *Modified Dijkstra* algorithm takes as input the graph $G_k = (V, L_k)$, the shortest path spanning tree T_k of G_k, and a link $(i,j) \in T_k, L_k$ and produces a spanning tree T_{k+1} of $G_{k+1} = (V, L_{k+1})$, where $L_{k+1} = L_k \setminus \{(i,j)\}$.

(1) The set of nodes V is partitioned into two sets: V_1 is the set of nodes whose shortest path from the root node on T_k includes link (i,j), and V_2 is the set of all remaining nodes.

(2) The modified Dijkstra assumes that shortest paths for nodes in V_2 are the same as the ones constructed in tree T_k. Thus, shortest paths for this set of nodes are not re-calculated.

(3) Calculates the shortest paths for the set of nodes in V_1 according to Dijkstra

Proposition 6.3: The tree produced by the Modified Dijkstra algorithm is a shortest path spanning tree.
Proof Follows from lemma 6.2. $\blacksquare$
The above modification of the Dijkstra algorithm is used to accelerate the updating of trees in the IAPR scheme in iterations $k \geq 1$.

B. Pre-processing on the complete graph

In order to accelerate the performance of the algorithm the following pre-processing step can be implemented. In graph $G(V, L)$ of the wireless mesh network the set L of links is reduced by considering only links (i,j) that have $w_N(i,j) \leq N_{max}$, i.e., only links with less than N_{max} neighbors are considered (see section V-B).

C. Average slack value on the SINR constraint as a stopping criterion

The proposed scheme eliminates one link at every iteration and by proposition 6.1 this will increase the total aggregate power P_k at each iteration k. Hence, before launching the algorithm for iteration $k+1$, the slackness of the SINR thresholds is tested by equally distributing the expected increase in aggregate power to all links in the mesh network. The test below will give a rough indication of how many SINR thresholds could be violated.

To this end, an exponential moving average of the increase in aggregate power levels at each iteration is calculated, ΔP_k. We assume that the average increase ΔP_k is allocated equally among the $|V|-1$ transmitting links. Thus, the SINR constraints for $(i,j) \in T_k$ can be written as follows,

$$u_{ijt} = \frac{g_{ij}\left(p_{ijt} + \dfrac{\Delta P_k}{|V|-1}\right) + \Lambda(1 - x_{ijt})}{\displaystyle\sum_{(m,n)\neq(i,j)} g_{mj}\left(p_{mnt} + \dfrac{\Delta P_k}{|V|-1}\right) + W} \tag{26}$$

Then, the fraction of SINR constraints that violate the required threshold is measured,

$$\varepsilon = \frac{\displaystyle\sum_{(i,j)\in T_k} \sum_{t=1}^{M} \mathcal{J}(u_{ijt})}{|V|-1} \tag{27}$$

where $\mathcal{J}(x)$ is the indicator function, defined in this case as,

$$\mathcal{J}(x) = \begin{cases} 1 & \text{if} \quad x < \gamma \\ 0 & \quad\ \ \text{otherwise} \end{cases} \tag{28}$$

if the value of ϵ is above a predefined threshold ϵ_T then the algorithm terminates.

D. Complexity of IAPR

The computational complexity that pertains one iteration of the algorithm is that of the modified Dijkstra algorithm, the pruning operation and the scheduling engine. Assuming a greedy packing heuristic for scheduling (see section VII), the complexity of each aforementioned step in one iteration is $\mathcal{O}(n \log n)$. In the worst case scenario, the algorithm terminates after K iterations, thus the complexity of the overall computational can be $\mathcal{O}(Kn \log n)$ steps.

VII. Numerical Investigations

In this section, we evaluate the performance of the proposed IAPR scheme compared to the MPR, MNR, IR and WPIR schemes that have been detailed in section V. Simulations are conducted on different randomly generated wireless mesh network topologies. For all different schemes a simple greedy heuristic for evaluating the scheduling has been used, which is described in algorithm 2. We denote by S the frame length achieved by either the optimal scheduling or the packing heuristic.

Algorithm 2 Packing Heuristic

Note: The packing heuristic does not perform power control and assumes that each link transmits with power that is 10% higher than the power needed to transmit on its own.

1: Let A be a list of all links sorted according to transmitted power (highest power first). Let B be an empty list and $t = 1$. At timeslot t schedule the first link in list A for transmission and shift it from list A to list B
2: **repeat**
3: Proceed down the current list A scheduling links for transmission in timeslot t, if feasible, and shifting them to list B if they transmit
4: Let $t = t + 1$
5: **until** A is empty
6: **return** $t - 1$

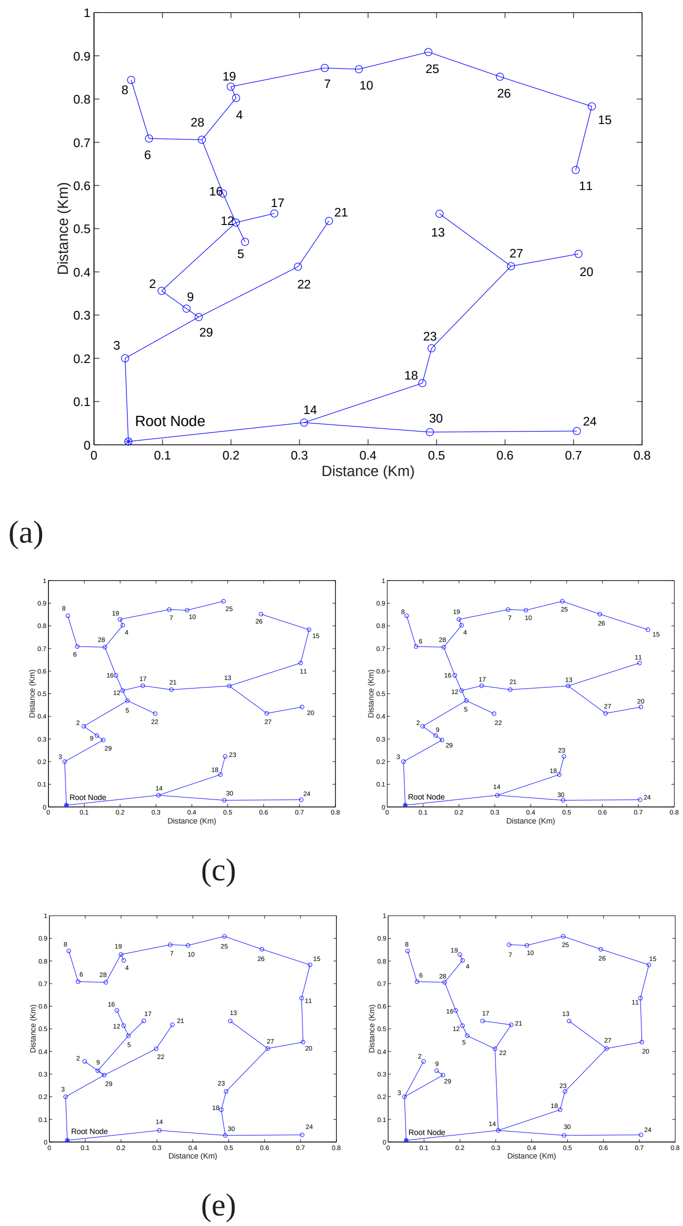
The Packing Heuristic tries to pack as many links as possible in each time slot that have not yet transmitted in previous time slots (list A), giving priority to the ones with the highest transmitted power. This continues until all links have transmitted at least once (list A is empty). This Packing Heuristic is similar to a heuristic used in [11], [12], [13], where it was shown to produce satisfactory solutions.

The IAPR scheme uses the packing heuristic at each iteration to evaluate the scheduling of the current shortest path spanning tree. Further, we use the following function to evaluate the interference caused by each link (i, j) in the shortest path spanning tree T_k: $I_k\left((i,j)\right) = w_N(i,j)$, i.e. the number of receiving nodes that are within the disc with center i and radius $d(i,j)$.

For the WPIR scheme the value of Θ has been selected to normalize the average power weight w_P and the average interference weight w_I. The value of $\beta = 0.5$ has been used in the following simulations that gives equal weight to the two metrics.

A. Results

Figure 5 shows the spanning tree constructed by the five different schemes in the case of a WMN with 30 nodes. The packing heuristic has been used for evaluating the scheduling in this scenario. The IAPR scheme requires 17 timeslots (5 (a)) compared to the MPR and the WPIR schemes, which both require 19 timeslots (5 (c) and (b) respectively). Thus, the IARP scheme provides 11% improvement on the minimum frame length. It is also interesting to note that for this scenario the MNR scheme achieved the same frame length as the IARP. Note also that even though the spatial reuse for MPR, WPIR and IR schemes is the same the constructed spanning trees are very different.



(a)

(b) (c)

(d) (e)

Fig. 5. Spanning trees constructed for the five different schemes with the assumptions being: 30 nodes (including the root node shown by star) uniformly distributed in a 1x1 km plane, path loss exponent is 3, γ = 5 dB, (a) Interference Aware Pruning Routing - IAPR (15 pruning operations) (S=17), (b) Weighted Power and Interference Routing - WPIR ($\beta = 0.5$) (S=19), (c) Minimum Power Routing - MPR (S=19), (d) Minimum Neighbors Routing - MNR (S=17), and (e) Interference Routing - IR (S=19).

The optimal joint power and routing problem, $OSR(L)$ (defined in III-B) has been solved for the small WMN with 18 nodes. For the specific topology, the minimum number of timeslots computed by CPLEX was 5 (this solution was found within 200 seconds). In figure 6 we compare the number of timeslots computed for the same topology for the different routing schemes with optimal and heuristic scheduling. As can be seen from the figure, when using the heuristic scheduling the pruning scheme provides a 6.7% improvement compared to the other routing schemes. It is also worth mentioning that when using the optimal scheduling, three out of five routing schemes achieve the same number of timeslots as the optimal joint scheduling and routing.

In the next set of experiment, 230 random uniformly distributed topologies of 40 nodes in a 3×3 km rectangular have been generated. The path loss exponent was assumed to be 4 and $\gamma = 5$. Figures 7, 8, 9,

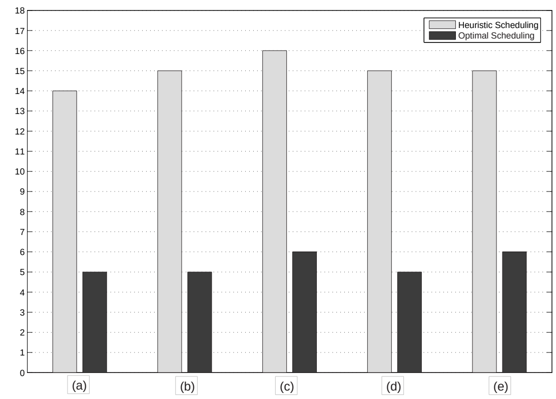


Fig. 6. Comparison between packing heuristic and optimal scheduling for the different routing schemes for the case of 18 nodes, (a) Interference Aware Pruning Routing (IAPR), (b) Minimum Power Routing (MPR), (c) Minimum Neighbors based Routing (MNR), (d) Interference Routing (IR) and (e) Weighted Power and Interference Routing (WPIR)

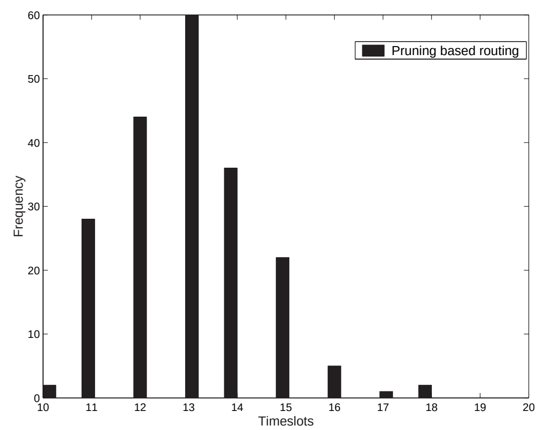


Fig. 7. Histogram of required frame length (in terms of timelots) for the Interference Aware Pruning Routing (IAPR) scheme

10, 11 shows the histogram of the frame length that has been achieved by the different routing schemes. We can infer from the histograms that the IAPR scheme achieves the best performance in comparison with the other routing schemes. Looking at the tail of the distributions the IARP scheme has 30% of frame lengths with timeslots greater or equal to 14, whereas the other schemes perform as follows: MPR 42%, MNR 70%, IR 71% and WPIR 57%.

The performance of the different routing schemes has also been tested with varying number of nodes in the WMN. The average frame length (in terms of timeslots) and the standard deviation of the frame length has been measured for 100 random uniformly distributed WMN's with 40, 60 and 80 nodes. The packing heuristic was used to evaluate the scheduling and the results are detailed in table I. From table I two interesting conclusions can be drawn. The first one is that in all different scenarios the IAPR scheme outperforms all the other routing algorithms. The average performance gains in terms of minimum frame length with respect to the second best routing scheme, which is MPR, range between 3.2% to 4.7%. It is interesting to note that the standard deviation of the IAPR averaged across all different scenarios is 13% better than that of the MPR scheme. This is of significant importance because not only the IARP scheme has better average minimum frame length but is also more robust to WMN topologies. Secondly, and as mentioned earlier, the MPR scheme outperforms the other routing schemes, except IAPR. This result reveals also the inherent difficulties of tuning the β, Θ values for the WPIR so that it can outperform the

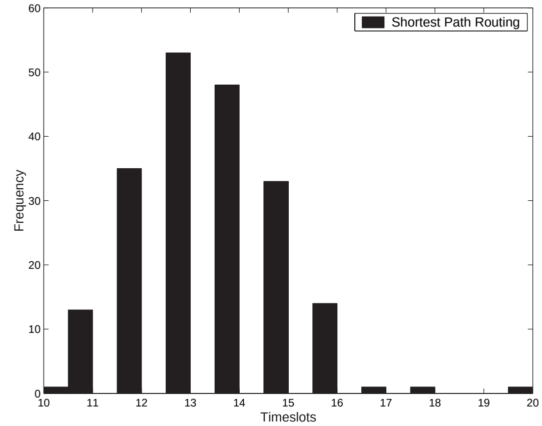


Fig. 8. Histogram of required frame length (in terms of timelots) for Minimum Power Routing (MPR) shceme

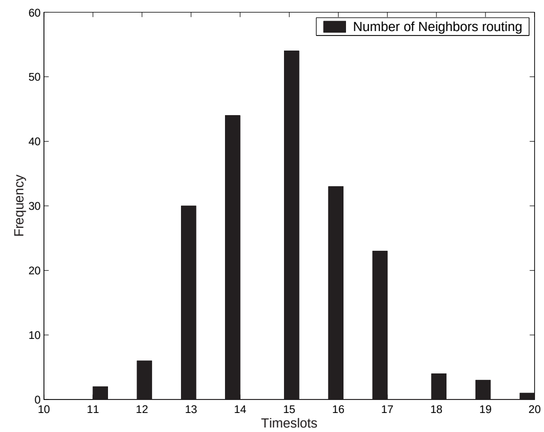


Fig. 9. Histogram of required frame length (in terms of timelots) for the Minimum Neighbors Routing (MNR) scheme

MPR.

For the same random topologies of 40 and 60 nodes, we test the performance of the different routing schemes using optimal scheduling and the results are shown in figure 12. When using the optimal scheduling the average frame length for all the different routing schemes is approximately the same (except for MNR and IR schemes which require slightly larger frame lengths). In other words, an optimal scheduling engine can compensate the decisions from the routing engine and thus being able to successfully pack all transmission in almost the same number of timeslots irrespectively of the routing scheme. Despite this fact, the IARP scheme is still very robust to different topologies. As can be seen from the error bar, which express the standard deviation, in figure 12, the std of the frame length for the IARP scheme is approximately 30% less than that of the MPR scheme.

One interesting question is that if we run the pruning algorithm for a fixed number of iterations, at which iteration will it find the schedule with the minimum possible frame length. To shed some light on that question, we have performed the following experiment. For a specific number of nodes in the network, namely 40 in this case, 100 uniformly distributed topologies have been generated in a 3×3km rectangular area. For each topology we perform 30 pruning operations and store the iteration where the IAPR scheme found the frame length with the minimum number of timeslots. We have repeated this procedure for each topology and the result is shown in figure 13, which depicts the empirical cumulative distribution function (cdf) that has been obtained by the experiment. The empirical cdf reveals that with 90% probability the pruning algorithm finds the schedule with the minimum timeslot span in less than 14 iterations.

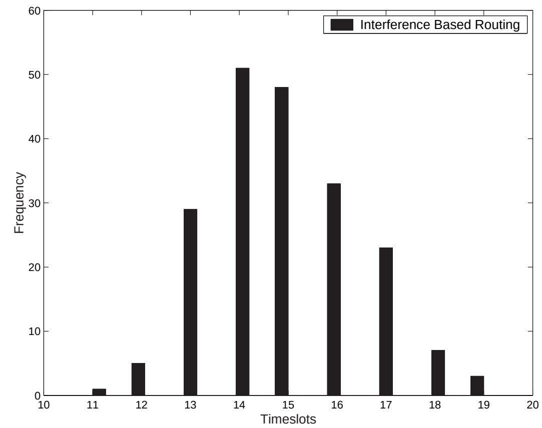


Fig. 10. Histogram of required frame length (in terms of timelots) for the Interference Routing (IR) scheme

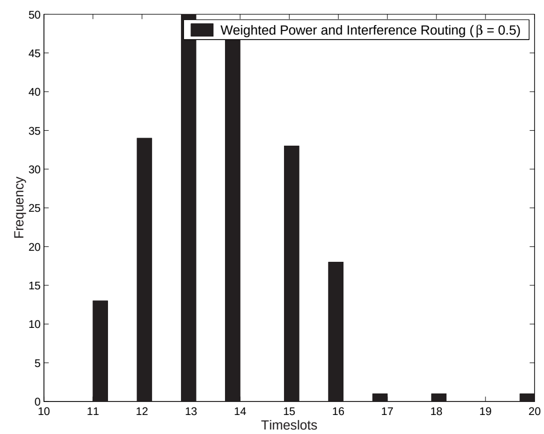


Fig. 11. Histogram of required frame length (in terms of timelots) for the Weighted Power and Interference Routing (WPIR) scheme

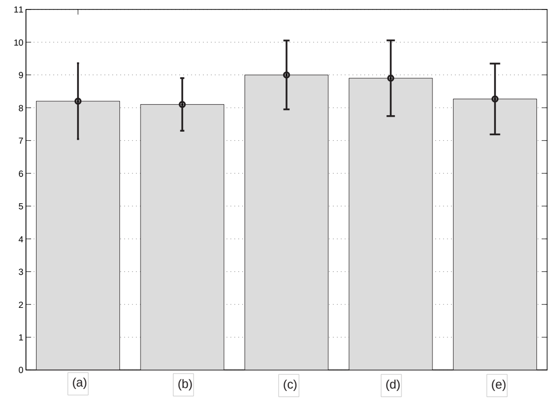
VIII. Conclusions

In this paper an interference aware pruning based routing scheme has been proposed that jointly optimizes path selection and STDMA scheduling. We have formulated the corresponding mixed integer program to perform joint scheduling and routing, and used this formulation to compared the performance of the proposed scheme with the optimal joint scheduling and routing. The routing schemes were evaluated using both a greedy scheduling heuristic and optimal scheduling. Extensive performance evaluation in different network settings of the proposed scheme revealed that it outperforms previously proposed routing schemes where interference and power consumption used as a routing metric.

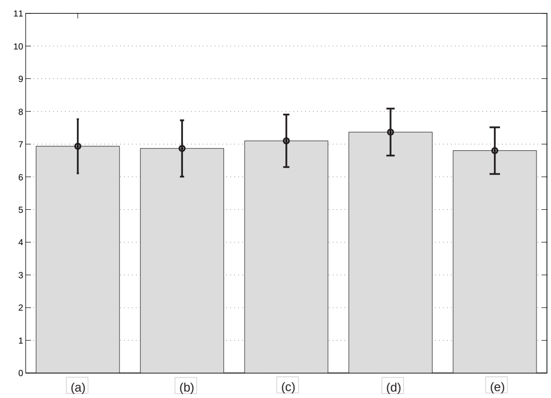
We should note that the general nature of the algorithms presented in this paper, including the proposed one, are applicable to more general network settings, that can include directional pattern transmissions and link gains that capture more precisely slow signal variations due to the physical terrain. We merely looked at the joint routing and scheduling problem without considering flows and the corresponding requirement of conserving them. This simplification allowed us to shed light into the interplay between the two engines and compare different schemes. We are currently working on such augmented models to incorporate network flow requirements.

References

[1] Ian F. Akyildiz, Xudong Wang, Weilin Wang, Wireless mesh networks: a survey, *Computer Networks, vol. 47, no. 4, pp. 445-487, March 2005*



(i)



(ii)

Fig. 12. Required number of timeslots for optimal scheduling in the case of (i) 60 nodes and (ii) 40 nodes: (a) Minimum Power Routing (MPR), (b) Interference Aware Pruning Routing (IAPR) , (c) Minimum Neighbors Routing (MNR), (d) Interference Routing (IR) and (e) Weighted Power and Interference Routing (WPIR)

[2] R. Nelson and L. Kleinrock, Spatial-TDMA: A collision-free multihop channel access protocol *IEEE Transactions on Communications, vol. 33, no. 9, pp. 934-944, September 1985*

[3] B. Hajek, G. Sasaki, Link scheduling in polynomial time, *IEEE Transactions on Information Theory, vol. 34, pp. 910-917, 1988*

[4] C. G. Prohazka, Decoupling link scheduling constraints in multihop packet radio networks, *IEEE Transactions on Computers, March 1989*

[5] A.M. Chou, V. O. Li, Slot allocation strategies for TDMA protocols in multihop packet radio networks, *IEEE INFOCOM ′92, 1992*

[6] Arash Behzad, Izhak Rubin, On the Performance of Graph-based Scheduling Algorithms for Packet Radio Networks, *IEEE GLOBECOM, San Francisco, CA, December 2003*

[7] S. Even, O. Goldreich, S. Moran, P. Tong, On the NP-completeness of certain network testing problems, *Networks, vol. 14, pages 1-24, 1984*

[8] L. Tassiulas, A. Ephremides. Stability properties of constrained queueing systems and scheduling policies for maximum throughput in multihop radio networks. *IEEE Transactions on Automatic Control, vol. 37, no. 12, pp. 1936-1948, December 1992*

[9] A. Das, R. Marks, P. Arabshahi, A. Gray, Power Controlled Minimum Frame Lenght Sheduling in TDMA Wireless Networks with Sectored Antennas, *IEEE INFOCOM'05, Miami, March, 2005*

[10] Geert Heijenk, Fei Liu, Interference-based routing in multi-hop wireless infrastructures, Computer Communications, vol. 29, pp. 2693 - 2701, 2006

[11] Gronkvist J. Traffic controlled spatial reuse TDMA in multi-hop radio networks. *PIMRC 1998; 1203-07.*

[12] Papadaki, K., and Friderikos, V.: 'Approximate dynamic programming for link scheduling in wireless mesh networks', Computers & Operations Research, 2007, doi:10.1016/j.cor.2007.02.010 .

[13] Peter Varbrand Patrik Bjorklund and Di Yuan, Resource optimization of Spatial TDMA in ad hoc radio networks: A column generation approach,*IEEE INFOCOM03, San Francisco, March 2003*

[14] R. L. Cruz, A. V. Santhanam. Optimal routing, link scheduling and power control in multihop wireless, networks, *IEEE INFOCOM03, San Francisco, March 2003*

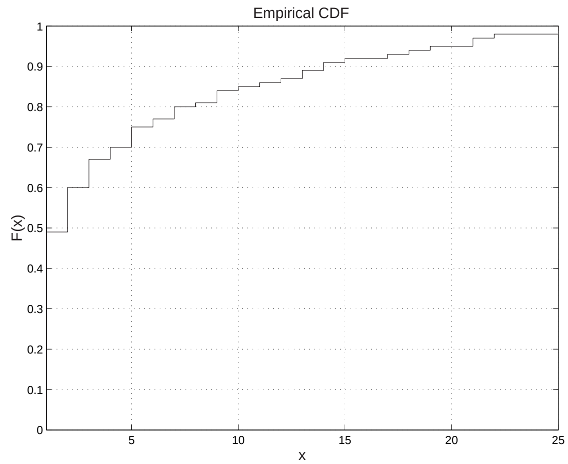


Fig. 13. The empirical cumulative distribution function (cdf) of the number of pruning iterations for finding the minimum schedule in terms of timeslots.

Nodes	40		60		80	
Timeslots	***avg.***	***std***	***avg.***	***std***	***avg.***	***std***
MPR	18.70	1.98	22.33	1.81	24.07	1.71
IAPR	18.10	1.97	21.27	1.57	23.10	1.37
MNR	20.27	1.95	23.8	1.86	26.63	1.63
IR	20.10	1.58	23.6	1.81	26.2	1.84
WPIR	18.93	1.78	22.43	1.55	24.26	1.82

TABLE I

PERFORMANCE OF THE ROUTING SCHEMES IN WMN TOPOLOGIES WITH VARYING NUMBER OF NODES

[15] T. V. Koushik Kar, Murali Kodialam, Leandros Tasiulas, Routing for network capacity maximization in energy-constrained ad-hoc networks, *IEEE INFOCOM03, San Francisco, March 2003*

[16] L. Chen, S.H. Low, M. Chiang, J.C. Doyle, Cross-layer congestion control, routing and scheduling design in ad hoc wireless networks, *IEEE INFOCOM, Barcelona, April 2006*

[17] M. Kodialam, T. Nandagopal, Characterizing Achievable Rates in Multi-hop Wireless MEsh Networks With Orthogonal Channels, *IEEE/ACM Transactions on Networking, vol. 13, no. 4, August 2005*

[18] Z. Wang and J. Crowcroft, Quality-of-service routing for supporting multimedia applications, *IEEE JSAC, vol. 14, no. 7, pp. 1228-1234, September, 1996*

[19] G. Liu, K.G. Ramakrishnan, A∗Prune: an algorithm for finding K shortest paths subject to multiple constraints, *IEEE INFOCOM, 2001*

[20] H. Takagi, L. Kleinrock, Optimal transmission ranges for randomly distributed packet radio terminals, *IEEE Transactions on Communications, vol. 32, no. 3, pp. 246-257, March 1984*

[21] Young-Jin Kim, Ramesh Govindan, Brad Karp, Scott Shenker, Geographic routing made practical, *USENIX Symposium on Network Systems Design and Implementation, Boston, Massachusetts, USA, May 2005*

[22] B. Gavish, Formulations and algorithms for the capacitated minimal directed tree problem, *Journal of the ACM, vol. 30, pp. 118-132, 1983*