**TDMA scheduling in long-distance wifi networks**

The MIT Faculty has made this article openly available. Please share how this access benefits you. Your story matters.

<table>
<thead>
<tr>
<th>Citation</th>
<th>Panigrahi, D., and B. Raman. “TDMA Scheduling in Long-Distance WiFi Networks.” INFOCOM 2009, IEEE. 2009. 2931-2935. © 2009, IEEE</th>
</tr>
</thead>
<tbody>
<tr>
<td>As Published</td>
<td><a href="http://dx.doi.org/10.1109/INFCOM.2009.5062261">http://dx.doi.org/10.1109/INFCOM.2009.5062261</a></td>
</tr>
<tr>
<td>Publisher</td>
<td>Institute of Electrical and Electronics Engineers</td>
</tr>
<tr>
<td>Version</td>
<td>Final published version</td>
</tr>
<tr>
<td>Accessed</td>
<td>Sat Dec 22 04:21:51 EST 2018</td>
</tr>
<tr>
<td>Citable Link</td>
<td><a href="http://hdl.handle.net/1721.1/58810">http://hdl.handle.net/1721.1/58810</a></td>
</tr>
<tr>
<td>Terms of Use</td>
<td>Article is made available in accordance with the publisher's policy and may be subject to US copyright law. Please refer to the publisher's site for terms of use.</td>
</tr>
<tr>
<td>Detailed Terms</td>
<td></td>
</tr>
</tbody>
</table>
TDMA Scheduling in Long-Distance WiFi Networks

Debmalya Panigrahi  
Massachusetts Institute of Technology  
debmalya@mit.edu

Bhaskaran Raman  
Indian Institute of Technology Bombay  
br@cse.iitb.ac.in

Abstract—In the last few years, long-distance WiFi networks have been used to provide Internet connectivity in rural areas. The strong requirement to support real-time applications in these settings leads us to consider TDMA link scheduling. In this paper, we consider the FRACTEL architecture for long-distance mesh networks. We propose a novel angular interference model, which is not only practical, but also makes the problem of TDMA scheduling tractable. We then consider delay-bounded scheduling and present an algorithm which uses at most 1/3rd more time-slots than the optimal number of slots without the delay bound. Our evaluation on various network topologies shows that the algorithm is practical, and more efficient in practice than its worst-case bound.

I. INTRODUCTION

In the recent past, long-distance WiFi networks have been proposed as a cost-effective means to provide Internet connectivity to remote rural regions [1]. Several deployments of such networks have been built around the world [2], [3]. Experience with such networks has shown that real-time video-conferencing based applications are very important, especially in developing regions of the world [3], [4].

We choose a TDMA-based approach since it allows the possibility of providing delay guarantees. TDMA scheduling with maximal spatial reuse has been extensively considered in past literature: e.g. [5], [6]. Here the goal is to minimize the TDMA schedule length, so as to maximize throughput. Optimal or even approximate TDMA scheduling in this context is a well known hard problem, closely related to graph vertex coloring [7], [6].

This paper considers the FRACTEL (WiFi-based Rural data ACcess and TELephony) architecture for long-distance mesh networks [8]. We propose and substantiate a novel angular threshold interference model in such networks; this model arises from a pattern of spatial reuse absent in generic mesh networks.

Under this interference model, we consider delay-bounded scheduling. We present an algorithm (Sec. III) which achieves the smallest possible delay bound, while using at most 1/3rd more time-slots than the optimal number of slots without the delay bound. We evaluate the algorithm on various topologies and find that it in fact performs better than its worst case bound in practice (Sec. IV)

Our work differs from the vast prior literature on TDMA scheduling in multi-hop wireless networks, in terms of the consideration of long-distance networks, and the unique angular threshold based interference model. Also, our algorithm considering a delay bound, and producing a schedule within a small, bounded factor of the optimal solution, is novel. Closely related to our work is [10], which also considers long-distance networks. But the model they consider is restricted in terms of using only highly directional antennas and only point-to-point links. The work in [11] is like ours in its consideration of path delay as a metric. But unlike in [11], (a) we consider the novel angular threshold interference model, (b) we consider a strict bound on the delay, and (c) our algorithm has a worst case bound on the TDMA schedule length.

II. PROBLEM SETUP

Long-distance links: A generic mesh network consists of links of various lengths. FRACTEL makes an architectural distinction between long-distance links and local-access links [8]. The distinction stems from two reasons: (a) long-distance links are formed using high-rise towers at one or both ends of a link. This is required to achieve Fresnel zone clearance above any obstructions [12] (see Fig. 1). And (b) each long-distance link typically uses a high-gain directional antenna at one end and a sector antenna at the other end [8]. This is required for achieving the long range. A typical setting is to have one central high-rise tower form long-distance links to several lower towers (Fig. 1); this amortises the high cost of the central tower [13].

Long-distance network (LDN) structure: The long-distance network (LDN), consisting of the long-distance links, is used to extend connectivity from a point of wired connection to various remote regions around it. In this paper, we shall consider only the LDN for TDMA scheduling. Now, since the long-distance links use high-gain directional antennas for the uplink (i.e. toward the central node or gateway) [13], [8]. This is again for achieving the range required for setting up the link. The uplink connectivity from a node (i.e. its parent)
is hence fixed, and determined during the planned network setup. In other words, the LDN topology is a tree.

Furthermore, with reasonable tower heights (40-50m), we can reach 20-25km with one hop easily in most flat terrains [13]. This implies that we can extend connectivity from the central node around a radius of 40-50km, with two-hops from the centre. Now, this covers a significant number of practical scenarios (although, arguably, not all). For instance, in India, each district has an optical fiber dropout at which the central node can be housed; and most districts are within 60-80km in dimension [14]. Hence a two-hop LDN is sufficient to cover most villages around the district headquarters. So we consider two-hop topologies in this paper. The ideas presented however, can potentially be extended to trees of greater depth as well. □

**Definition 1.** _The root node of the LDN is the central node which has wired connectivity. Children of the root node are known as hop-1 nodes; they are connected to the root via hop-1 links. Children of the hop-1 nodes are known as hop-2 nodes; they are connected to their respective hop-1 parents via hop-2 links._

**Traffic and channel models:** Most traffic flows in a rural mesh network is between leaves of the network and the root (or via the root to the Internet) [3]. Thus as an input to our TDMA scheduling, we assume that we are given the uplink as well as downlink traffic demands (in bits/sec) for each hop-2 node in the network.

For ease of exposition, we assume that the traffic demand to each hop-2 node is the same: say one unit, along the downlink direction (from the root node). And we assume that there is only one orthogonal channel (or frequency) available. [9] describes how our algorithm can easily accommodate relaxation of these assumptions. □

**Hop-1 node-splitting/link-replication, Network spoke model:** Given our input graph of a two-hop tree, we first make a simple transformation. We split each hop-1 node/link into as many parts as the number of children the hop-1 node has. Each of the replicated hop-1 node is attached to exactly one of the original hop-1 node’s child (hop-2) node. For example, in Fig. 2, original hop-1 node $X_1$ has three children $G_1$, $G_2$, and $G_3$. $X_1$ is split into three nodes $H_1$, $H_2$, and $H_3$; and each $H_i$ is connected to $G_i$, $i = 1, 2, 3$. The original hop-1 link $R-X_1$ is split into three hop-1 links: $R-H_i$, $i = 1, 2, 3$. The physical location of nodes $H_i$ is deemed to be the same as that of the original $X_i$ (although Fig. 2 shows them slightly away from one another, for clarity).

**Definition 2.** _After such transformation, a spoke of the graph is a hop-1 link and its corresponding hop-2 link. (Note that our spokes can be “bent” at the hop-1 node._)

For TDMA scheduling, each spoke now has a downlink traffic demand of one unit. We denote the total number of spokes of the (transformed) star graph by $n$; thus the graph has $2n + 1$ vertices and $2n$ edges. We denote the root node as $R$, the hop-1 nodes as $H_i$, the hop-1 links as $h_i$, the hop-2 nodes as $G_i$, and the hop-2 links as $g_i$, $i = 1..n$. These notations are indicated in Fig. 2. We denote the spoke consisting of $h_i$ and $g_i$ as $s_i$, $i = 1..n$. When we denote a link $l$ (without any index), $s_l$ denotes the spoke containing $l$. □

**Angular threshold interference model:** We consider link scheduling and hence a link interference model. Although a link can be scheduled in either direction, our interference model considers undirected links. We use the notation $x$ $\sim$ $y$ to indicate that links $x$ and $y$ do not interfere with one another, and the notation $x$ $\nsim$ $y$ to indicate that $x$ and $y$ mutually interfere.

In any interference model, a pair of adjacent links interfere with one another. This is reasonable since all the links at a node are setup atop the same central tower (many links may in fact use the same sector antenna in a point-to-multipoint configuration). Thus in our spoke model, $h_i \nsim h_j, \forall i, j,$ and $g_i \nsim g_j, \forall i$.

We first make an observation, unique to long-distance links. In Fig. 1, the further apart the two towers are, larger would be the Fresnel radius $r$, as well as the effect of earth’s curvature ($x$ in Fig. 1). Effectively, for a given pair of towers, of certain heights, there is a maximum distance $d$ beyond which they cannot “see” each other: due to lack of Fresnel clearance and earth’s curvature. This means that the radios on such towers will not interfere with one another either.

![Fig. 2. Hop-1 node-splitting/link-replication, Network spokes](image)

**TABLE I**

<table>
<thead>
<tr>
<th>Height of tower 1 (m)</th>
<th>Height of tower 2 (m)</th>
<th>Max. link length possible (d) (km)</th>
</tr>
</thead>
<tbody>
<tr>
<td>90</td>
<td>45</td>
<td>30</td>
</tr>
<tr>
<td>45</td>
<td>90</td>
<td>30</td>
</tr>
<tr>
<td>45</td>
<td>45</td>
<td>32</td>
</tr>
<tr>
<td>45</td>
<td>32</td>
<td>24</td>
</tr>
<tr>
<td>32</td>
<td>45</td>
<td>18</td>
</tr>
<tr>
<td>32</td>
<td>32</td>
<td>15</td>
</tr>
<tr>
<td>24</td>
<td>24</td>
<td>10</td>
</tr>
<tr>
<td>18</td>
<td>18</td>
<td>5</td>
</tr>
</tbody>
</table>

Table I tabulates the maximum distance $d$ for various values of the heights of the towers. For computing this table, we have taken an obstruction height of 12m (trees), and ignored land undulations. Such conditions are true in most plain (non-hilly) rural regions [13].

Now, intuitively, the larger the angle between $h_i$ and $h_j$, further apart are the towers at $H_i$ and $H_j$, or $G_i$ and $G_j$, and $H_i$ and $G_j$. Hence it is less likely that $g_i$ interferes with $h_j$ or with $g_j$. Our interference model captures this intuition in terms of an angular threshold $\theta_{thr}$. We denote the angle subtended at $R$ by $h_i$ and $h_j$ as $\theta_{ij}$ (note that only hop-1 links are involved in $\theta_{ij}$). In the _angular threshold interference model_, a hop-2 link $g_i$ interferes with both $h_j$ and $g_j$ if and only if $\theta_{ij} \leq \theta_{thr}$.

Note that $g_i \nsim h_j \Leftrightarrow g_i \nsim g_j$. For ease of exposition, we say $s_i \nsim s_j$ whenever $g_i \nsim g_j$. This is with the implicit
understanding that \( h_i \sim h_j \forall i, j \). Extending this notion, for a spoke \( s \) and a link \( l \), we say \( s \sim l \) whenever \( s \sim s_j \).

In this model, \( \theta_{hr} \) is an input parameter to our algorithms. We find in our evaluation on various topologies (Sec. IV) that in most cases \( \theta_{hr} < 30^\circ \).\)

**Summary:** To summarize, the input for our algorithm is the transformed graph. The scheduling problem maps directly to a link coloring problem, where interfering links ought to have different colors (i.e. time-slots in the TDMA schedule). Once we have a TDMA schedule for the transformed graph, the TDMA schedule for the original graph is obvious: the hop-1 links which were split into \( H_i, i = 1..k \) would be allocated the union of the time-slots given by the algorithm for the various \( H_i \).\]

### III. TDMA Scheduling in an LDN

We now present our algorithm for scheduling with strict delay restrictions. For a downlink flow, the delay incurred is the number of time units spent by a data unit (packet) at the intermediate hop-1 node. In a TDMA schedule, the time-slots are repeated in a cyclical fashion (see Fig. 3). In this scheduling cycle, if \( g_i \) gets a time-slot right after \( h_i \), the delay incurred is minimized: the delay is in fact zero. This is what we shall achieve in our algorithm. The algorithm seeks to minimize \( SL \), the schedule length (i.e. maximize throughput efficiency), subject to the above delay constraint. We present a sequence of four ideas that leads to our algorithm

![Fig. 3. TDMA scheduling cycle](image)

**Idea I:** Consider a sub-part of the network shown in Fig. 4(a), where we have a pair of mutually non-interfering spokes named \( s_1 \) and \( s_2 \). Now, we can color \( h_1 \leftarrow 1, g_1, h_2 \leftarrow 2, \) and \( g_2 \leftarrow 3 \). The fact that \( s_1 \sim s_2 \) allows us to use the same color for \( g_1 \) and \( g_2 \).

Now, in the network, we need at least two colors for the two spokes, whereas we have used three colors above. So the **overhead** in coloring just using this idea is 3.

**Idea II:** We improve upon I by considering 2 pairs of mutually non-interfering spokes: \( s_1 \sim s_2 \), and \( s_3 \sim s_4 \). This is shown in Fig. 4(b). We use the following lemma (refer to [9] for the proof).

**Lemma 1.** If \( (s_1,s_2) \) and \( (t_1,t_2) \) be two pairs of spokes where each pair is non-interfering, then there always exists a non-interfering pair \( (s_i,t_j) \), \( i,j \in \{1,2\} \), if \( \theta_{hr} < \pi/2 \).

![Fig. 4. Ideas used in our algorithm](image)

Without loss of generality, say \( s_2 \sim s_3 \), as indicated in Fig. 4(b). Now, we can color: \( h_1 \leftarrow 1, g_1, h_2 \leftarrow 2, g_2, h_3 \leftarrow 3, g_3, h_4 \leftarrow 4, \) and \( g_4 \leftarrow 5 \). As in I, we have reused colors across \( g_i \) and \( h_j \) when \( s_i \sim s_j \).

Now, for the four spokes in Fig. 4(b), we need at least 4 colors, whereas we have used five above. So the **overhead** in using II in a sub-part of the network is at most 3/4.

**Idea III:** Consider a situation where \( s_1 \) and \( s_2 \) are mutually non-interfering and \( s_3 \) is a spoke which does not interfere with at least one of \( s_1 \) or \( s_2 \). In the example in Fig. 4(c), \( s_3 \sim s_2 \).

We can now allocate colors as: \( h_1 \leftarrow 1, g_1, h_2 \leftarrow 2, g_2, h_3 \leftarrow 3, g_3, h_4 \leftarrow 4, g_4, h_5 \leftarrow 4, \) and \( g_5 \leftarrow 5 \). We have now used six colors where potentially only five are required (five spokes). So the **overhead** of using III is 4/3.

**Idea IV:** Our final idea is an improvement upon I, where we consider the existence of another pair of mutually non-interfering spokes \( s_4 \sim s_5 \) such that \( s_3 \) does not interfere with at least one of \( s_4 \) or \( s_5 \). In the example in Fig. 4(d), \( s_3 \sim s_4 \).

We can now allocate colors as: \( h_1 \leftarrow 1, g_1, h_2 \leftarrow 2, g_2, h_3 \leftarrow 3, g_3, h_4 \leftarrow 4, g_4, h_5 \leftarrow 4, \) and \( g_5 \leftarrow 6 \). We have now used six colors where potentially only five are required (five spokes). So the **overhead** of using IV, is 6/5.

Our algorithm is summarized in Algorithm 1. Given the above four ideas, the algorithm can be explained easily.

Clearly, the best ideas, in terms of the least overhead, are I and II. In both these, note that we consider pairs of mutually non-interfering spokes. So our algorithm first finds as many mutually non-interfering spokes as possible: Step-1.

If the above matching process leaves no unmatched spoke, there is no room for applying III or IV. But if there are unmatched spokes (set denoted \( S \)), we seek to associate each link of \( S \) with a pair of mutually non-interfering spokes, so that we can apply III or IV. We seek to apply IV first (case 2a in Algorithm 1), and apply III whenever this is not possible (case 2b in Algorithm 1).

In Step-3, we seek to apply II. If any pair of mutually non-interfering spoke remains after this, we seek to apply I, in Step-4. Even after this, if any spokes remain, we simply color using none of the above four ideas: this is Step-5.

A couple of remarks need to be made about the algorithm. Our first claim is that in Step-2, we can indeed ensure that if one link of a spoke in \( S \) is matched in \( M_2 \), then it is the hop-1 link. This claim is a direct consequence of the fact that the pair of links corresponding to a spoke in \( S \) are connected to the same set of vertices in \( B_1 \). Therefore, even if \( S_2 \) matches the hop-2 link in a spoke in \( S \), but does not match the corresponding hop-1 link, we simply change the end-point of this matching edge from the hop-2 link to the hop-1 link to obtain another maximum matching. This can also be ensured more elegantly using a minimum cost maximum flow to compute the maximum matching after suitably augmenting the bipartite graph. We omit details for brevity.

Our second claim is about Step-3. We claim that for any two pairs of spokes in \( B_1 \), there must necessarily be one
Algorithm 1 A scheduling algorithm satisfying delay bound of 0

Step-1: Find a maximum matching $M_1$ of the spokes where non-interfering pairs of spokes can be matched

Step-2: Let $S$ be the set of spokes unmatched. $B_1$ be the set of pairs of spokes matched under $M_1$ and $B_2$ be the set of hop-1 and hop-2 links corresponding to spokes in $S$

Construct a bipartite graph $B$ comprising vertex sets $B_1$ and $B_2$, where a vertex $u \in B_1$ is connected by an edge to a vertex $v \in B_2$ if and only if at least one spoke in the pair $u$ is non-interfering with $v$

Find a maximum matching $M_2$ in $B$ ensuring that in $M_2$, if only one link of a spoke in $S$ is matched, then that link is the hop-1 link

for each matched pair $(u, v) \in M$ where $v \in B_2$ is a hop-1 link do

Find a spoke among those corresponding to $u$ which is non-interfering with $v$; call this spoke $s_1$

Let $s_2$ be the other spoke corresponding to $u$ (and $s_3$ that corresponding to $v$)

Assign color $i+1$ to $s_2$, $i+2$ to $g_2, h_1$ and $i+3$ to $v, g_1$

$M = M \cup \{u, v\}$

Delete $u, v$ from $M_2$

Let $w$ be the hop-2 link in $s_1$

if $w$ is matched to some vertex $x$ in $M_2$ then

Case 2a

Let $s_3$ and $s_4$ be the spokes corresponding to $x$; let $s_3$ and $s_4$ be mutually non-interfering

Assign color $i+1$ to $w, h_3, i+2$ to $g_3, h_4$ and $i+3$ to $g_4$

Delete $w, x$ from $M_2$

else

Case 2b

Assign color $i+1$ to $w$

end if

end for

Step-3: Arbitrarily pair up vertices in $B_1$ that were not matched in $M_2$; call this pairing $M_3$

for each pair $(u, v) \in M_3$ do

Find a non-interfering pair of spokes, one corresponding to $u$ and the other corresponding to $v$; call them $s_1$ and $s_2$ respectively; let $s_2$ and $s_4$ be the other spoke of $u$ and $v$ respectively

Assign color $i+1$ to $h_2, i+2$ to $g_2, h_1, i+3$ to $g_1, h_2$ and $i+4$ to $g_3, h_4$ and $i+5$ to $g_4$

Delete $u, v$ from $M_3$

$i = i + 5$

end for

Step-4:

If there is a node $u \in B_1$ which has not been colored yet then

Let $s_1, s_2$ be the spokes corresponding to $u$

Assign color $i+1$ to $h_1, i+2$ to $g_1, h_2$ and $i+3$ to $g_2$

end if

Step-5: Pair up links in $B_2$ which have not been colored yet according to the spoke they belong to; call this pairing $M_4$

for each $(u, v) \in M_4$ where $u$ is the hop-1 link do

Assign color $i+1$ to $u$ and $i+2$ to $v$

Delete $(u, v)$ from $M_4$

$i = i + 2$

end for

spoke in each pair which are non-interfering. This follows as a direct consequence of Lemma 1. Thus, for any pairing at the beginning of Step-3, we can always find the pair of non-interfering spokes as required.

Analysis: The algorithm has a time complexity of $O(n^{2.5})$. Details appear in [9]. Further, the following theorem shows that the solution obtained by the algorithm is always close to the optimal solution (refer to [9] for a proof).

Theorem 1. Algorithm 1 uses at most $\lfloor (4/3) \times OPT \rfloor$ colors, where OPT is the number of colors used by any optimal channel allocation algorithm, and satisfies the property that a hop-2 link is scheduled immediately after its corresponding hop-1 link.

IV. PERFORMANCE EVALUATION

A. Evaluation setup

We generated 20 topologies each consisting of 20, 50, 100, 200 and 400 nodes, distributed randomly in a circular region of radius 40km on the x-y plane. For the purpose of our evaluation, we use simple mechanisms for the tower height assignment and the topology construction. We assume a tower of 45m height at the root node as well as at the hop-1 nodes; and a 20m tower at the hop-2 nodes.

To construct the topology, given the root node, we first label all the nodes within a threshold distance of the root as the hop-1 nodes and the corresponding links as hop-1 links. To choose this threshold distance, we refer to Table I (Ht1=Ht2=45m); this gives us a maximum link length of 27km for the hop-1 links. On the conservative side, we choose 25km to be the above threshold distance.

As a subsequent step, for each of the remaining nodes $X$, we locate the nearest hop-1 node, $X_{hop1}$. If dist$(X, X_{hop1}) < dist(X, R)$, then we label $X$ as a hop-2 node, with its parent as $X_{hop1}$. Else, we label $X$ as a hop-1 node. As a final step, for each hop-1 leaf node $Y_l$, we see if there is a hop-1 non-leaf node $Y_{nl}$ within 17km of it (refer Table I: Ht1=45m, Ht2=20m). If so, we make $Y_{nl}$ to be $Y_l$’s parent; $Y_l$ thus becomes a hop-2 leaf, which effectively reduces the tower
height requirement at that node from 45m to 20m.

Of course, before invoking the algorithm, we do the graph transformation step described in Sec. II.

B. Angular threshold interference model: $\theta_{thr}$ in practice

Our algorithm analysis assumes that $\theta_{thr} < 90^\circ$ [9]. We first checked if this is indeed valid on the various random topologies. For this, we assumed the hop-1 and hop-2 node heights as mentioned above, and we pessimistically assumed that RF interference could extend 5km beyond the maximum link length given in Table I. We further assumed a uniform radio transmission power of 20dBm, and that the required SIR (Signal-to-Interference Ratio) is 15dB (well above the theoretical value of 10dB for 11Mbps 802.11b transmission [15]). We considered three different kinds of antennas: a 24dBi parabolic grid, a 15dBi yagi antenna, and a 17dBi 90° sector antenna. We approximated the radiation patterns of these antennas from the vendor specifications as given at www.hyperlinktech.com. Furthermore, at each node, we used the antenna assignment algorithm as given in [13].

We considered the 20 different 400-node random topologies, and with the above settings, computed the threshold angle beyond which no pair of spokes mutually interfere. For 18 of the 20 topologies, this angular threshold was less than 30°; in one case it was 35° and in another it was 76°. This validates our assumption of $\theta_{thr} < 90^\circ$.

C. Results for algorithm performance evaluation

We plot the SL metric for networks of various sizes, averaged over the 20 random topologies for each network size (the standard deviation was very small and we do not report it). This is shown in Fig. 5. We show the results for various values of $\theta_{thr}$. We compare SL with $n$ since the latter is a lower bound on the schedule length.

We note that algorithm uses not more than 20-25% more colors than the lower bound of $\frac{3}{2}$ extra colors. We also observe from Fig. 5 that the performance is not very sensitive to $\theta_{thr}$, which is a nice property to have in practice.\footnote{Further evaluations, on real topologies as well as on random topologies with a high degree of asymmetry, are in [9].}

V. CONCLUSION

Wireless mesh networks using long-distance links have unique properties in terms of the spatial reuse pattern. In this paper, we have considered TDMA scheduling in long-distance networks. We have proposed an angular threshold interference model for such networks. Our algorithm lays emphasis on real-time applications, and considers delay minimization as one of the criteria. It produces a schedule which is at most 2/3 times longer than the optimal solution. Our evaluations show that in practice the solution performs better than the worst case bound.

The problem of optimal TDMA scheduling is a well known hard problem. Our work is novel in that we solve the problem with provable bounds, under the unique angular threshold interference model for long-distance wireless networks. Although we have considered the problem in the context of long-distance networks based on WiFi, our ideas can be applied for other wireless networks too, like WiMAX, so long as we have a network architecture with long-distance links.

ACKNOWLEDGMENT

This work was initiated when the authors were at Alcatel-Lucent Bell Labs, Bangalore, India. The FRACTEL project is supported in part by the Tata Telecom Centre, IIT Bombay.

REFERENCES