Revisiting MAC Design for an 802.11-based Mesh Network

Bhaskaran Raman
Department of CSE
Indian Institute of Technology, Kanpur
Kanpur, INDIA 208016
braman@cse.iitk.ac.in

Kameswari Chebrolu
Department of ECE
University of California at San Diego
La Jolla, USA 92093
chebrolu@cwc.ucsd.edu

Abstract: *This paper deals with an 802.11-based access network for rural villages. 802.11's CSMA/CA MAC is known to perform poorly in mesh networks. In this paper, we present the design of a novel MAC suited to a mesh network with outdoor, long-distance, point-to-point links. Multi-hop 802.11 networks are a topic of great interest currently, and our design represents a novel way to build such a network. Our MAC is a simple, 2-phase TDMA-based protocol (2-P). 2-P makes the wireless mesh network resemble a wired network closely – to the extent that all the links can operate simultaneously without mutual interference. 2-P is based on a novel flexibility in our network: Simultaneous Synchronous Operation (SynOp). We experimentally demonstrate SynOp on our field testbed. We discuss 2-P/SynOp's dependences, and their applicability in other multi-hop network scenarios.*

1. INTRODUCTION

The Internet revolution has *not* happened – at least from the perspective of the 90.2% of people [8] who have not yet used it. The goal of the Digital Gangetic Plains project[1] is to build a rural access network. *Low-cost* technology is essential for rural deployment since the density of users as well as their paying capacity is low [5, 11]. We have chosen 802.11 [3] as the underlying technology due to the cost benefits it offers [5].

The rural access network we envision is as represented by our testbed, depicted in Fig. 1. We have *outdoor*, *long-distance*, *point-to-point* links, constructed using high gain directional antennae. One or more nodes of the mesh network are *Landline-Access-Points (LAPs)*, which have wired Internet connectivity. We expect links to average about 5-10km in a real deployment (average inter-village distance), and we expect a LAP per district, with few tens of villages in it. We currently use the 802.11b variant of the technology.

While multi-hop 802.11 community networks [9, 6] are currently popular, our mesh network is significantly different from these. In terms of goal, our focus is on providing "points of connectivity", one at each village node, and not ubiquitous coverage (although our architecture can be extended to have last-hop omni-directional coverage as well). We expect the "points of connectivity" to bring in most of the initial benefits – applications such as distance learning, e-governance, market information access, VoIP, email, etc. in villages [11]. In terms of network architecture, we use directional antennae for our long-distance links. Directional antennae are also appropriate for our managed access network since these require some expertise to setup – these are generally avoided (at least by majority of the nodes) in a community network [6]. Also, in our mesh network, we use multiple radio devices per node, one for each directional link.

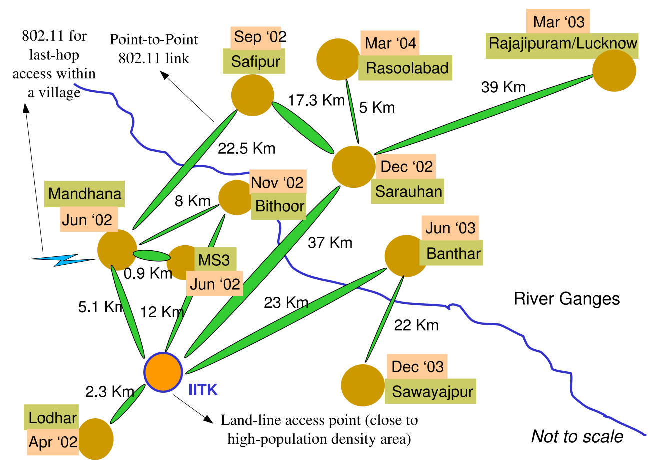

Figure 1: 802.11-based rural access network (our testbed)

Our design goal is to operate the mesh links with maximum performance, without mutual interference. A design point of focus in this paper is to operate under a single 802.11 channel, although there are up to three non-overlapping 802.11b channels. This has the advantage that if outdoor use of the channel is licensed[2], a network operator needs to license only one channel for operation in a region. Even in a license-free scenario, say in a community network, one could take the approach where the long-distance back-haul (outdoor) links use a single channel, and the others are used for regular WLAN (indoor) coverage at various points. Even if the long-distance links use more than one channel, our work is applicable in contiguous parts of the network which are operating under the

[1]Supported by Media Labs Asia, Project URL: http://www.cse.iitk.ac.in/users/braman/dgp.html

[2]This is the way it is currently in India; the govt. has however recently announced its intent to delicense the band "in-principle" for outdoor use.

same channel[3].

802.11 was fundamentally designed for *indoor*, *short distance*, *broadcast LANs* – in sharp contrast to our multi-hop, long-distance access network. Multi-hop CSMA/CA is known to be bandwidth inefficient [12]. This paper presents the design of an alternative MAC, suited to our network. The presence of directional links and multiple directional antennae at a node in our mesh network allows what we term as *Simultaneous Synchronous Operation (SynOp)*. In SynOp, each node either simultaneously receives or simultaneously transmits (independent data) along all its directional links (Sec. 2). We experimentally demonstrate that SynOp is practical despite the presence of side-lobes in the directional antennae.

We then present a simple 2-Phase TDMA-based MAC (termed *2-P*), which uses the SynOp flexibility effectively (Sec. 3). In 2-P, each node operates simply by simultaneously transmitting on all its links *(SynTx)*, right after simultaneously receiving from all its links *(SynRx)*. 2-P achieves maximum efficiency by being in one of these two modes at any time. An important feature of 2-P is that it does not require tight time synchronization. Further, the MAC modification required for 2-P can be implemented through firmware-level or even driver-level changes to off-the-shelf 802.11 equipment.

2-P/SynOp is a novel approach to a multi-hop 802.11 network. We avoid CSMA in the mesh – 2-P is a TDMA-based protocol. While TDMA-based schemes have been studied extensively in the past (see [10] and references therein), 2-P is novel in at least two important respects: (a) its use of the SynOp flexibility to keep all links active at all times, and (b) its operation without requiring tight time synchronization. With 2-P/SynOp, the wireless mesh now closely resembles a wired network in that all links can operate simultaneously without mutual interference. While the 2-P/SynOp design has emerged in the context of our rural access network, we believe that it also has applicability in other scenarios such as a planned wireless back-haul (e.g. for a community/campus network).

2. TOWARDS DESIGN OF A MAC

Before we design the MAC, we seek to understand the characteristics of the network architecture. In Sec. 2.1, we look at the performance of multi-hop CSMA/CA, under the use of multiple directional antennae per-node. We then motivate SynOp in Sec. 2.2, and experimentally demonstrate its feasibility in Sec. 2.3.

2.1 Performance of Multi-Hop 802.11 CSMA/CA

We now briefly illustrate the poor performance of 802.11 in a multi-hop setting. We focus on an aspect for which we could not find a ready reference among prior studies – performance using multiple directional antennae, and multiple radio interfaces per node. For comparison, we also consider a case with each node having *a single* omni-directional antenna. Call these *case-dirnl* and *case-omni* for brevity.

[3]It is very unlikely that the mesh is 3-edge colourable.

We consider UDP performance, without being concerned about the TCP state engine. For our study, we use the ns-2 simulator (v2.1b9a), with the following added features: (1) directional antenna support, and (2) presence of multiple wireless interfaces at a node. Tab. 1 compares the performance of various scenarios. In case-omni, we use a linear network, with all links in a straight line, and only adjacent nodes being able to hear each other. In case-dirnl, we use a similar configuration, but with adjacent links at 90^o to one another in the 2-hop and 3-hop cases. In all cases, we use static (manual) routing in ns-2, to discount the effect of any routing protocol, and have saturating CBR traffic flows (UDP packet size: 1.5KB) from one end of the network to the other. We turned off RTS/CTS (the results are similar even with RTS/CTS).

Antenna	1-hop (Mbps)	2-hop (Mbps)	3-hop (Mbps)
Dirnl.	6.1	2.8	2.7
Omni	6.1	3.0	2.0

Table 1: Performance of multi-hop 802.11

Tab. 1 presents the results. The throughputs in the 1-hop case are the base cases for comparison, and the values in this case are independent of the kind of antenna in use. In the 2-hop case (3 nodes), we observe that in case-omni, the throughput is half that of the 1-hop case. This is because the two links inherently cannot operate in parallel. In case-dirnl, the throughput actually comes down even below that of case-omni! The reason for this is as follows.

Suppose the network under consideration is $A-N-B$, with node N having two interfaces, each with a directional antenna facing in different directions (as in Fig. 2, with $\alpha = 90^o$). Denote the two interfaces at N as I_A and I_B respectively. Now, a property of directional antennae (which is also modeled in our version of ns-2), is that they have side-lobes and back-lobes – leakages along directions other than the main direction. Further, any directionality comes into picture only at larger distances, and it is irrelevant between interfaces I_A and I_B which are physically close-by, at N. That is, I_A can hear I_B's transmissions, and vice versa; of course, I_A cannot hear I_B's receptions (and vice versa).

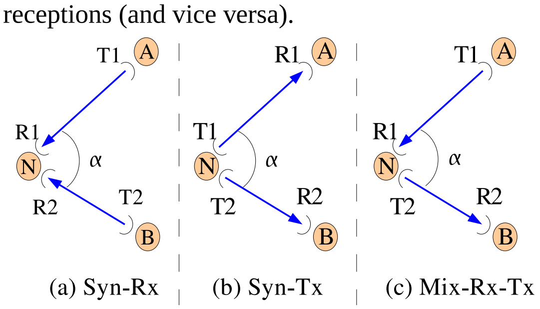

Figure 2: SynRx, SynTx, and Mix-Rx-Tx

Given this, the two links at N cannot operate *error-free* simultaneously, as we explain below. Now, operating simultaneously may mean one of three possibilities, as shown in Fig. 2(a),(b),(c): (a) node N receiving along both links *(SynRx)*, (b) node N transmitting along both links *(SynTx)*, and (c) node

N sending along one link while receiving along another *(Mix-Rx-Tx)*. (Note that the antennae are labeled differently in the three parts of the figure). In case(a), with immediate MAC-level ACKs, the ACK transmission of the packet which finishes reception first will interfere with the reception of the other interface - hence is not error free. In case(b), one interface of N will "carrier-sense" the other's transmission, and back-off as in the CSMA protocol – hence simultaneous transmission is not possible. In case(c), the interface which is receiving will face interference from the interface which is sending, and hence will suffer collision. Thus none of the three simultaneous operation cases are possible. With RTS/CTS, the only difference is that it will prevent packet errors, by preventing any simultaneous operation in cases(a)/(c).

This is the reason for the throughput-drop in the 2-hop scenario in case-dirnl. The throughput is even lesser than in case-omni, since the two interfaces at N now have additional carrier-sense induced back-offs.

In the 3-hop scenario, in case-omni, none of the three links can operate in parallel – although the two farthest links can operate in parallel, they do not due to the well known exposed-node problem (the middle two nodes are exposed to each other's transmissions to the respective farthest two nodes). This is the reason why the throughput is one-third of the 1-hop case. However, with directional antennae, the exposed node problem does not exist, and the throughput is more than one-third of the 1-hop case.

In more generic topologies, the performance of case-dirnl antennae would continue to be sub-optimal for the same reasons. And in case-omni, there is a possibility of higher interference between nodes in a general topology.

2.2 SynOp: Simultaneous Synchronous Operation

The simulations above clearly show that CSMA cannot operate the mesh network links in parallel. We are unable to take advantage of the presence of multiple directional antennae, and multiple interfaces per node. A straightforward, but crucial observation is that *arbitrary* contention resolution (using CSMA) is not required for the operation of the point-to-point links. Ideally, we seek to design a MAC which can operate the links at a node independently, without mutual interference.

Consider again, the simple scenario as in Fig. 2. Among the three possibilities, Mix-Rx-Tx is not feasible, under any MAC, since R_1 would experience too much interference from T_2's transmission (despite the directionality of the antennae). But, referring to our earlier discussion, SynRx was not possible due to the IFS-based *immediate* ACK mechanism, and SynTx was not possible due to the carrier-sense mechanism. SynRx and SynTx are potentially feasible without mutual interference among the two links – under an "appropriate" alternate MAC (which is the focus of this work).

We collectively term SynRx and SynTx as *SynOp: Simultaneous Synchronous Operation* – both links operating synchronously in the same direction. Even this is not straightforward and requires careful consideration, due to the presence of side-lobes, as we discuss below.

The radiation pattern of the parabolic grid antennae used in our testbed is shown in Fig. 3 [2]. We term the main direction gain minus (in dB terms) the side-lobe gain, as the side-lobe *rejection-level*, denoted SL. From Fig. 3, SL is about $25dB$ or more beyond about 10^o from the main direction. This pattern also applies to antenna sensitivity in the receiving direction, the overall gain being the sum of the gains at the transmitting and receiving antennae.

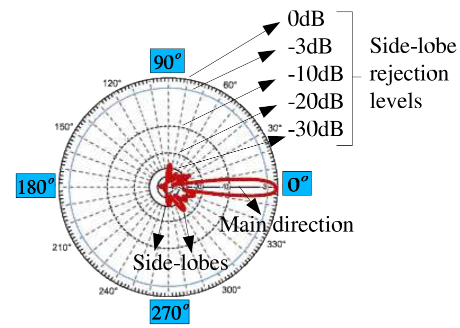

Figure 3: Spatial radn. pattern of parabolic grid antennae

To see how the presence of side-lobes may affect SynOp, consider SynRx – Fig. 2(a). T_2's transmission is seen as interference at R_1, after a side-lobe rejection of SL_α (rejection level at angle α between the two links). Let the path-loss from T_1 to R_1 be PL_1, and T_2 to R_1 (or R_2) be PL_2. Let P_{T_1} and P_{T_2} be the powers of transmission at T_1 and T_2 respectively. The received signal level at R_1 is thus $P_{R_1} = P_{T_1} - PL_1$; and the received interference level is $P_{T_2} - PL_2 - SL_\alpha = P_{R_2} - SL_\alpha$ (P_{R_2} is the received signal level at R_2 from T_2).

For "error-free" reception, we require the signal-level to be above the interference level by a certain amount. This is denoted SIR_{reqd} – the required signal to interference level[4]. This depends on the modulation being used, implementation losses, and also what the definition of "error-free" is – i.e., the required BER (bit-error-rate) level.

Thus we have, for error-free reception at R_1,

$$P_{R_1} - (P_{R_2} - SL_\alpha) \geq SIR_{reqd} \quad (1)$$

Similarly, for error-free reception at R_2, we have

$$P_{R_2} - (P_{R_1} - SL_\alpha) \geq SIR_{reqd} \quad (2)$$

For SynRx, we need mutual non-interference; combining Eq. 1 and Eq. 2 and rearranging, we get:

$$|P_{R_2} - P_{R_1}| \leq SL_\alpha - SIR_{reqd} \quad (3)$$

For SynTx (Fig. 2(b)), once again, R_1 would see interference from T_2 (and R_2 from T_1). Note that here the interference is felt at A (or B), and not at the middle node N. Proceeding as in the case of SynRx, we would finally end up with the following constraint, for SynTx to be feasible.

[4] This is actually the required SINR (signal to noise + interference ratio), but we ignore the noise level here; noise is usually below about $-110dBm$, and reception power much higher.

$$|P_{T_2} - P_{T_1}| \leq SL_\alpha - SIR_{reqd} \quad (4)$$

For SynOp, we require Eq. 3 and Eq. 4 to hold. This clearly depends on the angle of separation α between the two links, and importantly on SIR_{reqd}. For 11Mbps transmission in 802.11b, and a desired BER of 10^{-8}, the value of SIR_{reqd} can be theoretically estimated to be about $10dB$ [7]. Given this, and that SL_α is about $25dB$ for $\alpha > 10^o$ (see Fig. 3), the condition for SynRx reduces to

$$|P_{R_1} - P_{R_2}| \leq SL_\alpha - SIR_{reqd} = 25 - 10 = 15dB \quad (5)$$

This can be satisfied for a range of values of P_{R_1} and P_{R_2} (e.g. $P_{R_1} = P_{R_2}$). SynRx is thus feasible, and similarly, so is SynTx. The above calculation however does not account for many factors such as cable losses, RF leakages, and temporal variations in power levels. To determine the practicality of SynOp, we experimentally verify it in our testbed.

2.3 Experimental Verification of SynOp

Using off-the-shelf 802.11b hardware to experimentally verify SynOp presents a few complications. Referring to Fig. 2, we require all traffic (any packet sent, including ACKs) to be towards the common node N, to verify SynRx; and away from N to verify SynTx. This is difficult to ensure since the nodes are not synchronized – each node experiences random, independent medium-access delays. We do two things to address this. To achieve uni-directional traffic: (1) we turn-off RTS/CTS (which is anyway optional in off-the-shelf hardware), and (2) we use broadcast traffic from an AP device to a client device – this avoids MAC level ACKs.

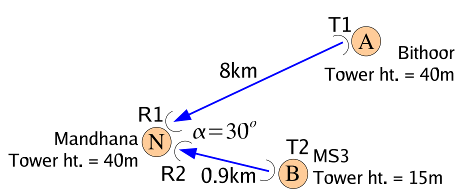

Figure 4: Experimental setup to verify SynRx

Our setup for verifying SynRx is shown in Fig. 4. We use three nodes of our testbed: *Mandhana*, *Bithoor*, and *MS3* in Fig. 1. The tower heights, link distances, and the angular separation between the links are as shown in Fig. 4. We use 802.11b (Cisco) Access-Points (APs) at the nodes A and B, and 802.11b clients (Orinoco) at the common-node N. We adjust the power levels at A and B such that the received power levels at N along the two links are about the same ($P_{R_1} \simeq P_{R_2}$). We have the two links set at the same 802.11b channel (we used channel 6). We have saturating UDP broadcast traffic (1.5KB packets at 1ms intervals) along the links $A \rightarrow N$ and/or $B \rightarrow N$, and measure the received throughput along each.

We observed that the throughput on each link with only that link active, was about 6.5Mbps. This value of 6.5Mbps is the maximum possible with a raw data rate of 11Mbps, after accounting for the PHY/MAC headers, and the slot-based carrier-sense mechanism.

With both links having traffic flooded too, the same throughput of about 6.5Mbps was observed on both links. The two links were thus able to receive simultaneously, without mutual interference. This shows that SynRx is feasible.

The situation is symmetric with respect to SynTx, but the setup for SynTx presented an additional problem. As pointed out in Sec. 2.1, SynTx does not really happen if one interface starts transmission and on hearing it, the other backs-off. Avoiding the back-off is a little tricky, and here we used a "feature" allowed by Cisco 350 series APs [1], which is what we used at the common node N. Each AP has two antennae connectors for diversity – and the driver allowed us to have it transmit through one of the connectors (say "left"), and receive on another (say "right"). In each AP, we connected (only) the "left" connector to the external high-gain antenna – while it was listening/carrier-sensing on the "right" connector, which of course sees only negligible amount of noise. With this setup, we were able to avoid the back-off and experimentally verified SynTx as well. We were able to achieve a throughput of 6.5Mbps simultaneously on both links.

So far our discussion has revolved around the simple topology given in Fig. 4. We discuss how the same can apply to the entire network in Sec. 4. For now, we assume that SynOp is feasible at all nodes in the network, and discuss the design of a MAC which builds on top of it.

3. 2-P: A MAC BASED ON SYNOP

SynOp allows us to potentially operate the links at a node simultaneously. We now discuss how to enable such operation throughout the network. We first describe *2-P*, a simple 2-phase TDMA-based MAC protocol built on SynOp, in Sec. 3.1. We then describe in Sec. 3.2 as to how 2-P can operate without tight time synchronization.

3.1 The 2-P MAC Protocol

The primary goal in the design of the MAC is bandwidth efficiency. Now, SynOp achieves maximal efficiency locally at a particular node, by having each link active (receiving or transmitting) at any time. The idea behind the MAC is essentially to enable SynOp at all nodes at all times. We achieve this as follows: (1) each node is either in SynRx mode, or in SynTx mode; if a node is in SynRx mode, its neighbors have to be in SynTx mode, and vice-versa, and (2) each node switches between SynRx and SynTx modes; and when a node switches, so do its neighbors. This simple TDMA-based mechanism is the basis of our MAC protocol. We term this protocol as *2-P* to indicate that it consists of each node switching between two *phases*: SynTx and SynRx.

In 2-P, we require adjacent nodes in the topology to be in different phases (SynRx/SynTx). Clearly, this is possible if and only if the topology has no odd-cycles – that is, bipartite. Nodes in different partitions would be in different phases.

This is depicted in Fig. 5. (V_1, V_2) is the bipartition of the network, and transmissions $V_1 \rightarrow V_2$, and $V_1 \leftarrow V_2$ are in alternate *phases*. We term the combination of two consecutive phases as a *round*. We shall revisit the bipartition constraint in Sec. 4, and for now assume that the topology is bipartite.

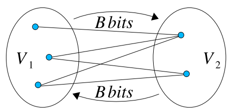

Figure 5: Two-phase scheduling in a bipartite network

The duration of each phase is fixed, for all the nodes in the network. For simplicity, we also discuss only the case where the two phases are of equal duration[5]. In the absence of data from higher layers, the 2-P MAC sends dummy bytes so that there is always data transmission during SynTx.

3.2 Loose Synchronization for the 2-P MAC

Synchronization is central to any TDMA system, and is required to ensure collision-free operation. In our setting, there are two kinds of collisions possible: (1) both ends of a link are transmitting simultaneously, and (2) a node is transmitting along one link, and receiving along another (Mix-Rx-Tx). One way to avoid collisions is to have tight time synchronization across all the nodes in the network. However, a system with tight synchronization is in general more difficult to design, engineer, and implement. Further, since the inter-node distances in our setting can be high, with several tens of μs one-way propagation delay, these have to be accounted for accurately in any time synchronization protocol.

We take an alternate approach that does away with the need for perfect time synchronization. The key insight is that both kinds of collisions can be avoided by purely local decisions, without requiring global time synchronization. Our scheme relies on three simple rules: (r1) In SynRx, each node waits for the end of transmission from *all* of its neighbors, (r2) then *immediately*, it begins transmission on all of its links to its neighbors simultaneously (SynTx), and (r3) it switches to SynRx *immediately* after transmission. It is easy to see that these conditions ensure that both kinds of collisions described above do not happen. And switching phases as soon as possible condition ensures that system idle time is minimized.

With this loose synchronization, note that it is not necessary that all nodes in the same partition of the graph are in the same phase (SynTx or SynRx). But this does not matter, since only local synchronization is required for SynOp. Also, we have implicitly assumed here that the multiple radios at a node are in tight communication with one another. This is of course possible to implement. The above is the base mechanism – we now outline how we handle various errors/failures.

We first note that the above mechanism is robust to packet CRC errors – a node can still detect the presence/absence of transmission from its neighbor(s). It can thus detect the end of such transmission from its neighbor(s) and switch to SynTx. A little additional mechanism is required however, to handle a case where a packet may be completely lost (no signal received). In such a case, the node which is expecting to receive cannot wait indefinitely. In fact, in 2-P, if one node were to wait indefinitely, all nodes eventually will be waiting to hear from their neighbors, in a deadlock.

To prevent indefinite waits, we have a simple *timeout* mechanism. On entering SynRx phase, a node starts a timer, of value denoted T_0. On expiry of the timer, if no *signal* has been detected from one or more of the neighbors, the node enters SynTx phase anyway. Of course, the timer is canceled as soon as a signal is detected from *all* the neighbors. The timer value used is the same at all the nodes.

Note that if a node times out waiting for a neighbor, we do not want it to enter SynTx before finishing reception from other neighbors. Hence it makes sense to choose a value for T_0 higher than the duration of a phase, denoted d_p – this is the time for which that node may be receiving from the other neighbors. (In our simulations summarized below, we chose T_0 to be $1.25 \times d_p$).

The final issue we discuss concerns arbitrary possibilities of simultaneous timeouts and packet errors. Temporary out-of-synchrony may cause two neighboring nodes to have overlapping SynTx phases (the first kind of collision we talked about earlier). A subtle but important point to note is that such a situation will get corrected immediately, in the next round, as we explain below.

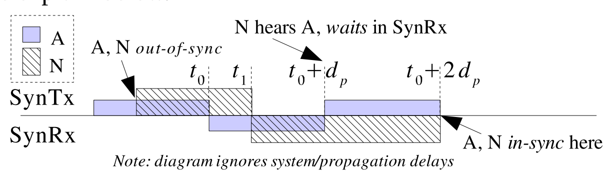

Figure 6: Self-synchronization in 2-P

Suppose the two neighbors are A and N whose SynTx phases are overlapping at this point in time. Without loss of generality, assume that A's SynTx phase finishes first, say at time t_0, and N's SynTx finishes at $t_1 > t_0$. A's SynRx is from t_0 to $t_0 + d_p$. Now, N would still be in SynRx at $t_0 + d_p$, when A switches back to SynTx (N would have been planning to switch to SynTx at $t_1 + d_p > t_0 + d_p$). Thus N would start to hear from A at $t_0 + d_p$, and *will wait* until it finishes hearing from A SynTx fully, before switching to SynTx itself (due to (r1) above). This waiting process ensures that the two nodes are now in-sync. Thus, within one round, the two nodes regain (loose) synchrony! 2-P is thus self-synchronizing. This is depicted in Fig. 6.

Above, there can be rare cases where the two nodes overlap in SynTx completely ($t_1 \simeq t_0$ modulo system/propagation delays). In such a case, both nodes will timeout for each other in the next round, and the cycle will repeat. This can however be avoided easily – if a node experiences repeated timeouts on a link, it can add a small *random* delay to its timeout, before entering SynTx. Since both the nodes will likely choose

[5]They could potentially be unequal to match the traffi c pattern on certain links – we do not consider this in this paper.

different random values, the deadlock will be broken, and the SynTx phases will no longer overlap completely.

In the above scheme, link failures are detected through continuous timeouts. And link recovery is no different from the loss of synchrony discussed above. This thus also covers the case where the system is coming up from an initial state.

While we are currently working on implementing 2-P for a thorough performance analysis, we have performed preliminary simulations. In our simulator, we have modeled: transmission and propagation delays, random system delays at each radio of a node, packet errors, packet losses, as well as node/link failure/recovery. We have verified that 2-P is indeed self synchronizing as described above – temporary loss of synchrony gets fixed in the immediate next round. This is so even in the presence of simultaneous node failure/recovery (as tested on up to 64-node district topologies – see below).

4. DISCUSSION AND CONCLUSIONS

We now briefly present some of the issues related to 2-P/SynOp which we are currently addressing. We also present other points of discussion.

Network Topology Issues: SynOp and 2-P are closely related to the network topology. For SynOp, the conditions Eq. 3 and Eq. 4 are dependent on link distances and inter-link angles. In addition, 2-P requires a bipartite topology. We summarize our two main findings so far here. (1) Given an arbitrary topology, it is possible to write a set of linear (in)equalities, with the transmission power along each link as variables – much like Eq. 3, 4. This set of equations represents the feasibility of SynOp throughout the network. (2) Given a set of (village) nodes, we have designed simple heuristics to construct a *bi-connected* bipartite topology for which the above set of linear equations is feasible. We have tested our algorithms on villages from Durg district, Chattisgarh, India, and have been able to construct feasible topologies of up to 64 nodes [4].

Higher Layer Issues: In 2-P, the duration of each phase can be fixed to be a multiple of the system MTU (Maximum Transfer Unit). Also, it makes sense to accommodate more than one MTU sized packet per round, to minimize RTT overhead (RTT for a 10km link $\simeq 67\mu s$; transmitting 1KB at 11Mbps $\simeq 700\mu s$). If higher-layer packets are smaller than the MTU size, in each phase, as many higher layer packets as possible can be sent, with dummy padding at the end if necessary. The link-layer can now use piggybacked-ACKs, instead of the IFS-based ACKs as used in 802.11.

Implementation Approaches: Since 2-P is a MAC modification to 802.11, we believe that it can be implemented with minimal modifications to 802.11, thus preserving the cost-benefits. There are two possible approaches, both of which we are exploring. First, 2-P can be implemented in a different firmware, interfaced to an 802.11 baseband processor. We are exploring this with Intersil's HFA 3863. The other, more flexible option is to implement driver-level changes to enable SynRx/SynTx, much like in our experimental setup. This would require faking IP unicast packets as MAC broadcast packets (and providing an alternate link-level ACK mechanism), and also disabling back-off as in our experiment. As of this writing, we have such an implementation based on the Linux Host-AP driver (http://hostap.epitest.fi), working for a single link.

Dependences of 2-P/SynOp and Wider Applicability: SynOp essentially allows us to reuse the same 802.11b channel for the multiple links at a node. There are two dependences here: (a) the static nature of the network, and (b) multiple radios at a node. 2-P/SynOp can thus be potentially used in other scenarios, such as a planned 802.11 wireless back-haul (e.g. for a community network). Also, the dependence on the PHY is minimal, and is related to (a) the SL_α as determined by the antenna design, and (b) SIR_{reqd}, as determined by the modulation. Both these parameters are captured in the equations in Sec. 2.2. The equations thus apply to any PHY, and not just 802.11, and the ideas are potentially applicable for other wireless technologies as well.

In summary, we have used 802.11 as a low-cost technology for building a rural access network. In this paper, we have presented the design of the 2-P MAC protocol for our mesh network. 2-P is based on the SynOp flexibility, which is enabled by the presence of multiple directional links per node. 2-P/SynOp represent a novel and efficient way to build a multi-hop 802.11 mesh network, quite different from existing multi-hop wireless networks.

Acknowledgment: We thank Media Labs Asia (MLA) for supporting this effort. The MLA team at IIT-Kanpur, Sreekanth Garigala, and A. R. Harish, have been instrumental in the outdoor experiments. The ns-2 simulator modifications were done by Ashwini Kumar and Sabyasachi Roy. We also thank K. Lakshminarayanan for his feedback on the paper.

5. REFERENCES

[1] Cisco a350 Data Sheet. `www.cisco.com/warp/public/cc/pd/witc/ao350ap/prodlit/a350a_ds.htm`.

[2] HyperGain HG2424G Reflector Grid Antenna. `http://www.hyperlinktech.com/web/hg2424g.php`.

[3] IEEE P802.11, The Working Group for Wireless LANs. `http://grouper.ieee.org/groups/802/11/`.

[4] Topology for Durg District, Chattisgarh, India. `http://www.cse.iitk.ac.in/~bhaskar/dgp/durg-topo/`.

[5] Pravin Bhagwat, Bhaskaran Raman, and Dheeraj Sanghi. Turning 802.11 Inside-Out. In *HotNets-II,* Nov 2003.

[6] S. Biswas and R. Morris. Opportunistic Routing in Multi-Hop Wireless Networks. In *HotNets-II*, Nov 2003.

[7] A. Chindapol, A. Stephens, and J Lansford. *IEEE P802.15-02/069r0, Change request for 802.15 Recommended Practice PHY text*.

[8] ITU. World Telecommunication Development Report. `http://www.itu.int/ITU-D/ict/publications/wtdr_03/`, Dec 2003.

[9] R. Karrer, A. Sabharwal, and E. Knightly. Enabling Large-scale Wireless Broadband: The Case for TAPs. In *HotNets-II*, Nov 2003.

[10] A. Sen and M. L. Huson. A New Model for Scheduling Packet Radio Networks. In *INFOCOM*, 1996.

[11] R. Tongia et al. Discussion notes. In *ICT for Sustainable Development, Bangalore Workshop*, Jan 2004.

[12] S. Xua and T. Saadawi. Revealing the problems with 802.11 Medium Access Control Protocol in Multi-Hop Wireless Ad Hoc Networks. *Computer Networks*, 38, 2002.