Reinforcement Learning System to Mitigate Small-Cell Interference Through Directionality

Anton Paatelma*†, Danh H. Nguyen†, Harri Saarnisaari*, Nagarajan Kandasamy, and Kapil R. Dandekar
* CWC, University of Oulu, Finland. Email: anton@paatelma.com, harri.saarnisaari@ee.oulu.fi
Drexel University, Philadelphia, PA. Email: {dnguyen, kandasamy, dandekar}@drexel.edu
† Co-primary Authors

***Abstract*—Beam-steering techniques using directional antennas are expected to play an important role in wireless network capacity expansion through ubiquitous small-cell deployment. However, integrating directional antennas into the existing wireless PHY and MAC stack of small cells has been challenging due to the added protocol overhead and lack of a robust antenna beam selection technique that can adapt well to environmental changes. This paper presents the design, implementation, and evaluation of LinkPursuit, a novel learning protocol for distributed antenna state selection in directional small-cell networks. LinkPursuit relies on reconfigurable antennas and a synchronous Time-Division Multiple Access (TDMA) MAC to achieve simultaneous directional transmission and reception. Further, the system employs a practical antenna selection protocol based on the well known adaptive pursuit algorithm from the reinforcement learning literature. We implement a real-time prototype of LinkPursuit on the WARP platform and conduct extensive experiments to evaluate its performance. The empirical results show that appropriate use of directionality in LinkPursuit can result in higher network sum rates than omnidirectional transmission under various degrees of cross-link interference.**

I. Introduction

Network densification—the practice of deploying more radio access nodes into a geographical area—is being considered as a cost and bandwidth-effective method to increase wireless network capacity. In a dense heterogeneous network, small-cell systems, (also known as *femtocells*) co-exist and cooperate with high-power macrocells to serve users' traffic demands. To mitigate the interference problem in such a dense deployment, many advanced management techniques have been proposed, including those that use antenna directionality [1]. By using directional antennas, small-cell network nodes can focus energy in only the intended direction, thereby creating less interference between links and more potential for spatial reuse. Nevertheless, bringing these techniques to practice has been challenging for two reasons: *(i)* the difficulty of integrating directional antennas into the existing wireless physical layer (PHY) and medium access control (MAC) stack of small cells, and *(ii)* the lack of robust antenna beam-steering (or beam selection) techniques that can cope well with the wireless channel's stochastic nature and dynamics in the operating environment of small cells.

978-1-5386-3531-5/17/$31.00 © 2017 IEEE

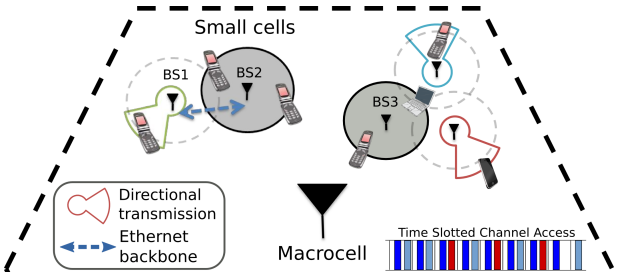


Fig. 1: System model of a small-cell network with high spatial reuse through antenna directionality. BS1-5 are co-channel small-cell base stations (BSs) serving multiple clients synchronously

In this paper we explore a low-overhead and efficient way to integrate directionality into wireless small-cell operations using compact, electronically reconfigurable antennas. Unlike beamforming techniques that employ multiple Radio Frequency (RF) chains to achieve directionality, reconfigurable antennas integrate multiple miniature radiating elements on a single device, thereby producing several distinct radiation patterns without any additional RF circuitry. This makes reconfigurable antennas immediately applicable for use with client-side mobile devices, which usually have a single RF chain and can use only one antenna at a time. To make small-cell interference management practical with reconfigurable antennas, we propose LinkPursuit, a novel wireless network design that deploys reinforcement learning-based antenna state selection methods on top of a synchronous MAC protocol and adapts well to environmental changes.

The challenge of designing efficient MAC protocols with directional antennas for wireless ad hoc networks has been well studied [2]. A substantial number of directional MAC protocols in the literature belong to the contention-based, random access category. Our work focuses on building a directional MAC protocol that coordinates contention-free directional transmissions and receptions to occur simultaneously in an optimal way. Along this line, DIRC [3] and MiDAS [4] systems also explore multi-beam adaptive antenna systems for simultaneous transmissions in the same time slot. However, unlike ours, these approaches rely heavily on protocol coordination to assess and schedule antenna directions and are usually not resilient to intermittent environmental changes.

Stochastic online learning using reinforcement learning algorithms has recently gained significant attention in

the wireless community [5, 6]. The body of work on applying multi-armed bandit (MAB) formulation in wireless communications is also rich, encompassing spectrum sensing and secondary user access [5], as well as antenna subset selection [6]. However, these studies are still primarily theoretical and simulation-based. An experimental study using MAB with real measured channels can be found in [7] for the problem of antenna state selection, but the learning process also appears to progress offline on channel traces. Furthermore, the proposed antenna selection policies assume a non-varying channel condition, which is not suitable for real-world deployment. In contrast, we focus on modifying a powerful reinforcement learning policy to operate in real time and adapt well to dynamic interference conditions.

Our contributions in this paper are three-fold: First, we present the LinkPursuit wireless architecture, which employs reconfigurable antennas and a TDMA MAC to achieve simultaneous directional transmission and reception for interference avoidance and spatial reuse. For antenna orientation, LinkPursuit relies on reinforcement learning to dynamically select in each time slot the optimal antenna states at both the transmitter and receiver link ends with negligible protocol overhead. The system model for LinkPursuit is shown in Fig. 1. Second, we propose a concrete and robust antenna state selection method for use in LinkPursuit. Unlike prior applications of reinforcement learning in this area, we formulate the antenna state selection task as a non-stationary MAB problem, wherein the reward generating processes associated with the bandit's arms undergo changes over time. Our solution uses a well-known selection policy called *adaptive pursuit* [8], which we carefully tailor to the distributed nature of a wireless link where decisions need to be split between the transmitter and receiver. At each decision-making epoch, a network node decides on an active antenna mode based on observations of the outcomes from previous choices, with the objective to identify the optimal antenna state which maximizes the Packet Delivery Ratio (PDR). Third, we implement LinkPursuit on the Wireless open-Access Research Platform (WARP) [9] and conduct a series of real-time over-the-air experiments indoor to quantify its performance with respect to both omnidirectional transmission and less practical antenna state selection schemes. The empirical results show that appropriate use of directionality in LinkPursuit can result in higher network sum rates in dense small-cell deployments, delivering on an average 70% increase in network-sum PDR over omnidirectional transmission under various degrees of interference.

The paper is organized as follows: Sec. II describes LinkPursuit system architecture. We present in details the antenna selection protocol of LinkPursuit in Sec. III. Sec. IV describes our experimental methods and results. Finally, we conclude the paper in Sec. V.

II. LinkPursuit System Architecture

In this section we present the wireless network design of LinkPursuit. The system constitutes a cross-layer design which enables the MAC layer to assume control of antenna orientations and schedule (select) the optimal antenna configuration it perceives. It requires only a single RF chain per radio node and is agnostic of the PHY signaling method used, which makes LinkPursuit applicable to both LTE small-cell and 802.11 networks. Below we highlight several design decisions and the rationale behind them.

A. Antenna Subsystem

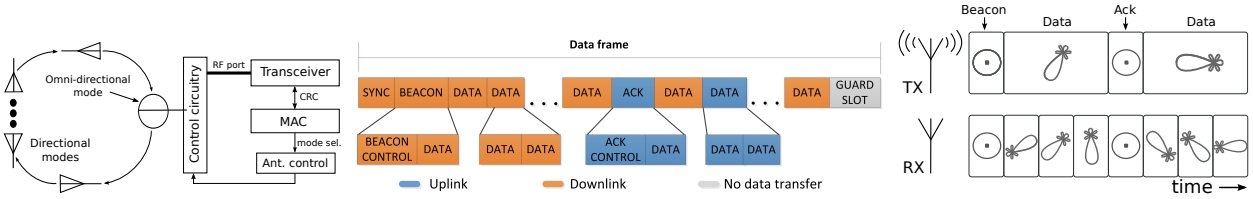
The antenna subsystem of LinkPursuit is depicted in Fig. 2a. First, *the design relies on compact reconfigurable antennas* to realize directional beams using a single RF chain. Unlike smart antenna systems which use multiple RF chains to achieve directionality through beamforming, reconfigurable antennas integrates multiple radiating elements on a single device, with a switchable impedance matching circuit for each supported configuration. This enables the antennas to produce steerable directed beams with lower processing overhead in a smaller form factor, making it highly suitable for power and cost-constrained client devices, such as smart phones and laptops. The drawback is that only one antenna state can be used or accessed at a given time.

Second, *the antenna subsystem provides a number of discrete antenna states*, including one omnidirectional mode and one or more directional modes. Wireless network devices use the omnidirectional antenna mode for idle listening, management packets, and control packets. They use one of the directional modes for data transmission and reception. These design choices are based on the following observations: *i)* control packets are often of broadcast nature, *ii)* directional transmission and reception improve network capacity, provided that suitable antenna orientations are used, and *iii)* omnidirectional antenna state is needed to maintain standard compliance in other traditional network settings.

Third, *antenna control is delegated to the MAC layer*, which configures the antenna state prior to pushing a packet into the PHY buffer for transmission. On the receiving end, the MAC software also selects an antenna state for reception in a given time slot. This packet-based antenna control eliminates the need to augment the physical layer to implement transmit antenna assessment and receive diversity selection logic. It also enables the LinkPursuit system to be immediately deployable on existing PHY chipsets via a firmware upgrade.

B. MAC Layer Design

LinkPursuit divides time into time slots and employs a TDMA-based MAC protocol to support antenna state assessment and scheduling in a network-wide setting. The MAC layer incorporates necessary mechanisms



(a) Reconfigurable antenna system **(b)** TDMA frame structure **(c)** MAC-assisted antenna steering

Fig. 2: LinkPursuit system architecture showing the interplay between the antenna subsystem and MAC protocol

for PHY control, time synchronization, time-frequency-space resource scheduling, packet buffering, and antenna selection logic. Here we highlight the important MAC features that directly support custom antenna selection policies, and refer the reader to [10] for a full description of our prototype implementation, including all the listed functional blocks.

TDMA Frame Structure: a TDMA frame constitutes a series of time slots for packet transmission. As depicted in Fig. 2b, it starts with a *Sync signal* from the main BS (e.g., macrocell), followed by a number of *Physical Resource Blocks (PRBs)*, and ends with a *Guard time slot*. Each PRB is further broken down into a number of regular time slots (shown as two in Fig. 2b), which can be used to send packets in different antenna orientations. The small-cell BSs assign available PRBs in the TDMA frame to their respective clients on a per-user per-link (up- or downlink) basis, and links in the same collision domain may be concurrently scheduled. However, the antenna orientations used in each Data time slot of a PRB by the scheduled sender and receiver are completely up to the antenna state selection logic, which we will discuss in details in Sec. III. Figure 2c illustrates the MAC and antenna operations of an example network link throughout the TDMA frame.

As designed, LinkPursuit MAC layer supports four resource block types: Beacon, Ack, Data, and Empty. Beacon PRBs are used by base stations to broadcast `BEACON` control packets which carry resource schedules, link statistics, and other control information from their service set (each BS and its served clients form a service set). Ack PRBs are used by clients to send `ACK` packets for block-acknowledging data reception in previous Data slots, as well as passing the clients' link states. Note that acknowledgements are not sent following each Data packet, but must be aggregated in a single control packet per TDMA frame for efficiency. In addition, control packets from different service sets that are sent in the same control time slot should be orthogonal either in time, frequency, or code to ensure correct data link operations. Data PRBs are used for data transfer in either up- or downlink directions, and an Empty PRB denotes an unallocated block. In our current implementation, each TDMA frame consists one Beacon, one Ack, and 16 Data PRBs. Each PRB has two time slots of 1.6 ms each. Under the reconfigurable antenna-equipped synchronous wireless system described above, we next describe a distributed antenna state selection policy based on the concept of reinforcement learning.

III. Antenna Selection Protocol

This section addresses the antenna state selection requirement at all scheduled wireless nodes in each data time slot. We cast the problem as a multi-armed bandit optimization, a specific class of reinforcement learning problems. MAB represents the classic trade-off between exploitation and exploration in that an agent, through repeated interactions with the unknown environment and analyzing the received stochastic rewards, must choose between *(i)* maximizing expected profit using current knowledge of the environment, achieved by selecting the currently perceived best arm (i.e., option), and *(ii)* trying to learn more about the environment by exploring other arms to improve the quality of its decisions. Many well-known MAB selection policies, such as the family of Upper Confidence Bound (UCB) policies [11], have been shown to achieve the optimal performance on the cummulative reward. However, these policies are developed for the stationary bandit problem, in which the reward generating processes are stochastic but stationary over time. Below, we consider a non-stationary MAB problem that is more suitable for a dynamic network setting and propose to adapt and integrate the *adaptive pursuit selection policy*, developed by Thierens [8], into the complete wireless MAC/PHY stack in LinkPursuit.

A. Problem Formulation

Consider a small-cell network with an arbitrary number of base station (BS) and client pairs operating in the same wireless channel. Each BS serves a client for whom it would like to maximize the downlink throughput over a finite number of time slots. At the beginning of each downlink time slot (uplink can be handled similarly), the BS and client are presented with M and N possible antenna states, respectively. The BS selects a state for packet transmission, and the client selects one for reception in that time slot. In MAB terms, the combination of the Tx antenna state at the BS and Rx antenna state at the client forms an $M \times N$ choice (or arm) matrix $\mathbf{A} = \begin{pmatrix} a_{11} & \cdots & a_{1N} \\ \vdots & \ddots & \vdots \\ a_{M1} & \cdots & a_{MN} \end{pmatrix}$. After making a choice in time slot t, the BS-client link receives a numerical reward $R(t)$ depending on the selected arm a_{ij} in that time slot.

In this work we use PDR as the reward metric and assume that the reward in time slot t for some selected antenna state a_{ij} is drawn from an unknown Bernoulli distribution: with probability $\mu_{ij}(t)$ the packet is received successfully, i.e. reward $R(t) = 1$; otherwise, $R(t) = 0$. Further, the reward distribution mean for each arm a_{ij} changes over time, that is, $\exists t_1, t_2 \in [1, T]$ and $t_1 \neq t_2$, such that $\mu_{ij}(t_1) \neq \mu_{ij}(t_2)$.

Our objective is to develop a robust antenna state selection strategy that can adapt well to the changes in the reward distributions while at the same time maximizing the expected reward output.

B. Adaptive Pursuit Method Overview

Algorithm 1 Adaptive Pursuit Selection Policy [8]

Input: $M, N, P_{min}, \alpha, \beta$
Output: $\{a_{\bar{i}\bar{j}}\}$ ▷ Series of antenna states to select
$P_{max} \leftarrow 1 - (M \cdot N - 1)P_{min}$; $t \leftarrow 0$
for $i \leftarrow 1$ **to** M **do** ▷ Initialize $\mathbf{P}$ and $\mathbf{Q}$ matrices
  for $j \leftarrow 1$ **to** N **do**
    $P_{ij}(0) \leftarrow \frac{1}{M \cdot N}$; $Q_{ij}(0) \leftarrow 1.0$
  end for
end for
while NOTTERMINATED( ) **do** ▷ Main loop
  $a_{\bar{i}\bar{j}} \leftarrow$ PROPORTIONALSELECTSTATE($\mathbf{P}(t)$)
  $R(t) \leftarrow$ GETREWARD($a_{\bar{i}\bar{j}}$) ▷ Update rewards
  $Q_{\bar{i}\bar{j}}(t+1) = (1-\alpha)Q_{\bar{i}\bar{j}}(t) + \alpha R(t)$
  $i^*j^* = \text{ARGMAX}_{ij}\,(\mathbf{Q}(t+1))$ ▷ Update prob. of sel.
  $P_{i^*j^*}(t+1) = P_{i^*j^*}(t) + \beta\,[P_{max} - P_{i^*j^*}(t)]$
  for $i \leftarrow 1$ **to** M **do**
    for $j \leftarrow 1$ **to** N **do**
      if $ij \neq i^*j^*$ **then**
        $P_{ij}(t+1) \leftarrow P_{ij}(t) + \beta\,[P_{min} - P_{ij}(t)]$
      end if
    end for
  end for
  $t \leftarrow t + 1$ ▷ Advance time index
end while

We propose to use the adaptive pursuit strategy due to its fast convergence toward the current optimal solution [8]. Originally proposed for learning automata, the adaptive pursuit strategy is a probabilistic selection policy; it identifies at each time step (slot) the optimal selection probability $P_{ij}(t)$ for every antenna state a_{ij} such that the expected cumulative reward is maximized at the end of the run. The arms' selection probabilities are specified in an *operator probability matrix* $\mathbf{P}(t) = \begin{pmatrix} P_{11}(t) & \cdots & P_{1N}(t) \\ \vdots & \ddots & \vdots \\ P_{M1}(t) & \cdots & P_{MN}(t) \end{pmatrix}$, where $0 \leq P_{ij}(t) \leq 1$ and $\sum_{i,j} P_{ij}(t) = 1$. Toward this reward maximization goal, the adaptive pursuit algorithm maintains an *operator quality matrix* $\mathbf{Q}(t) = \begin{pmatrix} Q_{11}(t) & \cdots & Q_{M1}(t) \\ \vdots & \ddots & \vdots \\ Q_{M1}(t) & \cdots & Q_{MN}(t) \end{pmatrix}$ that keeps a running estimate of the reward for each arm. Whenever arm (antenna state) a_{ij} is selected, its current reward estimate $Q_{ij}(t)$ is updated with the corresponding received reward $R(t)$ using an exponential, weighted averaging mechanism as:

$$Q_{ij}(t+1) = (1-\alpha)Q_{ij}(t) + \alpha R(t) \tag{1}$$

where the adaptation rate α, $0 < \alpha \leq 1$ discounts the past reward estimates obtained for arm a_{ij}.

At each time step t, the adaptive pursuit method *biases in its random selection* toward the operator $a_{i^*j^*}$ that currently has the maximum estimated reward $Q_{i^*j^*}(t)$, using a "winner take all" strategy: it increases the selection probability of the best arm toward P_{max}, while decreasing all other selection probabilities toward P_{min}, $0 < P_{min} < P_{max} < 1$. The selection probabilities for the next time slot are updated as follows:

$$\begin{aligned} P_{i^*j^*}(t+1) &= P_{i^*j^*}(t) + \beta\,[P_{max} - P_{i^*j^*}(t)]\,, \\ &\qquad \text{for } i^*j^* = \text{argmax}_{ij}\{\mathbf{Q}(t)\} \\ P_{ij}(t+1) &= P_{ij}(t) + \beta\,[P_{min} - P_{ij}(t)]\,,\ \forall ij \neq i^*j^* \end{aligned} \tag{2}$$

under the constraint $P_{max} = 1 - (MN - 1)P_{min}$. The learning rate β determines the convergence speed and accuracy, and the constraint ensures that if $\sum_{i,j} P_{ij}(t) = 1$, the sum of the updated selection probabilities equals one in the next time step. We summarize the adaptive pursuit policy for antenna state selection in Algorithm 1.

C. LinkPursuit Antenna Selection Protocol

LinkPursuit employs a practical modification of the adaptive pursuit selection policy presented above for its antenna selection logic. We alter the selection policy to be conducive to the distributed nature of a wireless link: decisions are split between transmitter and receiver and coordinated via a MAC protocol. In LinkPursuit, every node in the network maintains two link-state tables for each of its unicast links: a *Send table* to derive the optimal antenna state for packet transmission to its link counterpart and a *Receive table* to derive the optimal antenna state for packet reception from that node. For a downlink transmission, the BS will use its Send table, and the client will use its Receive table to select the optimal antenna states at both ends in any scheduled time slot.

Each Send and Receive table contains two matrices: an *antenna state selection probability matrix* $\mathbf{P}$, and an *antenna state quality matrix* $\mathbf{Q}$ (explained in Sec. III-B) which keeps track of the reward estimates for all available antenna state combinations. In a downlink scenario, entries in the BS Send table's matrix contain the pursuit statistics for arms a_{ij}, representing the combination of the BS's Tx antenna state i and the client's Rx antenna state j. Similarly, the BS Receive table, which is used to orient antennas in an uplink transmission, keeps pursuit statistics for arm a_{ji} which is the combination of the client's Tx antenna state j and BS's Rx antenna state i.

After synchronizing with the network, a node receives resource allocation schedules through Beacons and knows

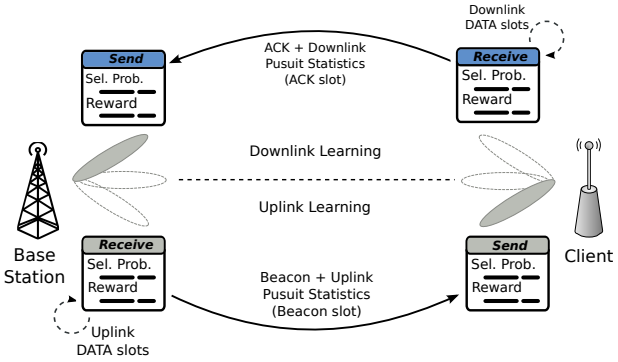


Fig. 3: Pursuit statistics tables at the BS and the client, and the update process. Tables that maintain identical statistics at opposite link ends have the same color

when it is supposed to receive a packet. This scheduling information forms the basis of our PDR reward metric: if a packet is not correctly received by a network node (i.e., failing CRC check) in a scheduled receive time slot, the node perceives a reward of 0 for that send-receive mode combination. Otherwise, it receives a reward of 1. Other major components of the antenna selection protocol are summarized below.

Updating Pursuit Statistics: Since our reward metric is the PDR, the actual reward outcome—whether a packet is received or not in each time slot—is observable only by the link receiver. Therefore, the Receive table at the receiver node gets updated continuously following each transmission. In contrast, the Send table at the transmitter node is updated only through the MAC protocol operations, when the receiver passes its most recent Receive table via a control packet (Beacon or Ack). Essentially the Send table at the transmitter is a cached copy of the Receive table maintained online at the receiver. For any unicast link, it is crucial to maintain coherency between its transmitter's Send table and receiver's Receive table by not letting them drift too far apart in time without periodic updates, as they independently supply the pursuit statistics for distributed selection of send antenna state at the transmitter and receive antenna state at the receiver. The pursuit statistics tables and their update processes are shown in Fig. 3.

Distributed Antenna State Selection: The adaptive pursuit algorithm assumes that in each time slot, both the BS and client jointly select an arm a_{ij}, the combination of Tx antenna state i and Rx antenna state j, from a joint probability distribution specified by the operator probability matrix $\mathbf{P}$. However, this is not feasible in practice due to the prohibitive overhead of maintaining an up-to-date $\mathbf{P}$ at both link ends and jointly performing the selection in each time slot. To reduce the coordination overhead, LinkPursuit divides the joint antenna state selection process into two separate phases, one for Tx and the other for Rx antenna selection, following the definition of conditional probabilities: $\Pr(S \cap R) = \Pr(S) \cdot \Pr(R|S)$, where S and R are random variables representing the Tx and Rx antenna states, respectively. As a result, the Tx antenna state in time slot t is selected randomly according to the *pursuit marginal distribution* of send modes $P^S(t)$, specified as:

$$P_i^S(t) = \Pr(S = i) = \sum_{j=1}^{N} P_{ij}(t),\ i = 1, \ldots, M \quad (3)$$

The Rx antenna state is then selected conditionally from the *pursuit conditional distribution* of receive modes, given a preselected sending mode i $(1 \leq i \leq M)$, specified as:

$$P_j^R(t) = \frac{\Pr(S = i \cap R = j)}{\Pr(S = i)} = \frac{P_{ij}(t)}{P_i^S(t)},\ j = 1, \ldots, N \quad (4)$$

Essentially, a unicast link transmitter preselects a random send antenna mode for *each scheduled PRB* of the link in the TDMA frame, according to the current statistics in its Send table. Note that the chosen send mode applies to all time slots in the PRB to keep the network-wide interference condition relatively static for that single PRB and enable the algorithm to converge. To make this information available at the receivers for Rx antenna selection, LinkPursuit delegates all Tx send mode selections to the base station, including both downlink and uplink data traffic in the TDMA frame. At the beginning of the frame, since the BS has all needed pursuit statistics in its Send table (for downlink send mode selections) and uplink Receive table (for uplink send mode selections on behalf of clients), it preselects send antenna modes for all transmissions scheduled in the frame and broadcasts this information together with the link schedule. On the other hand, the link receiver selects a receive mode dynamically in *each time slot* it is scheduled to receive, given the chosen send mode in that slot and using its up-to-date Receive table. This distributed antenna selection method is pursuant to the update frequency of pursuit statistics at both link ends (Send tables are updated once per TDMA frame, while Receive tables in each time slot).

IV. Implementation and Evaluation

We implement LinkPursuit on the WARP platform by augmenting its open-source *802.11 Reference Design* with our Reconfigurable Alford-Loop Antenna (RALA) [12] and custom software layers for the TDMA and antenna selection operations described in Secs. II and III. Our RALA can radiate in both an omnidirectional pattern as well as four directional beams at 90° separation. We add to the FPGA-based 802.11 PHY design a programmable antenna controller hardware, realized using a generic 16-pin GPIO header connecting to the antenna circuitry. The MAC software framework is implemented using a dual-core architecture with two embedded Microblaze soft processor cores. Our testbed implementation details can be found in [10]. The antenna selection procedure is implemented according to Fig. 4.

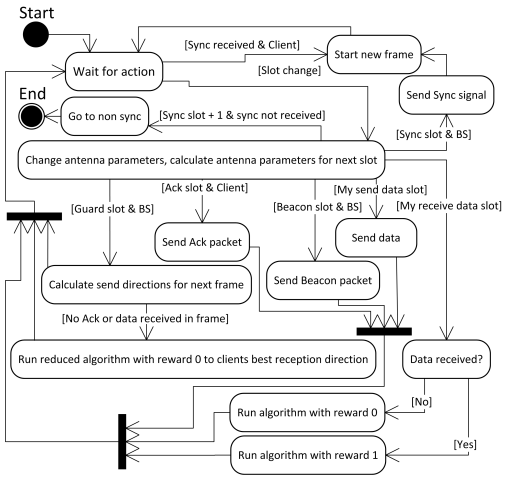


Fig. 4: Flow diagram of the LinkPursuit antenna adaptation process

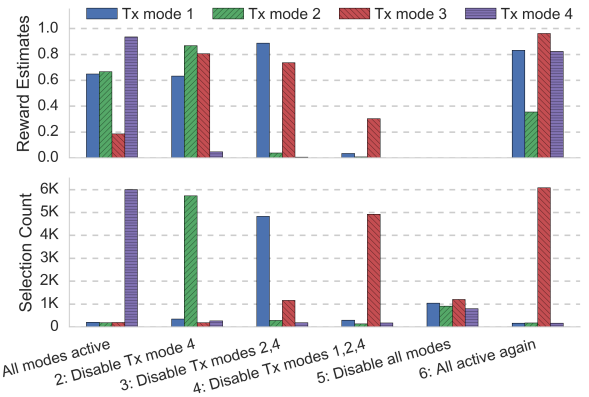


Fig. 5: Reward estimates (top plot) and the actual selection counts (bottom plot) for the four possible directional Tx modes under LinkPursuit's learning policy. Each verification step encompasses 6600 downlink packets.

We evaluate LinkPursuit's performance using real-time over-the-air transmissions under both a single-link scenario with no co-channel interference, and scenarios involving two concurrent interfering links. In our experiments presented below, we allocate all available PRBs for downlink transmission, with multiple BSs concurrently transmitting in each time slot. All control packets (Beacons and Acks) belonging to the same control time slot are further orthogonalized in time (ordered by service set), but Data packets are allowed to interfere. We conduct all experiments on WiFi channel 14 with fixed PHY settings at QPSK with convolutional code rate of 1/2, yielding a consistent PHY rate of 12 Mbps. The adaptive pursuit parameters are set by default to be $\alpha = 0.05, \beta = 0.1$, and $P_{max} = 0.9$. These parameters are chosen based on practical observations of the protocol performance.

A. Microbenchmark Verification

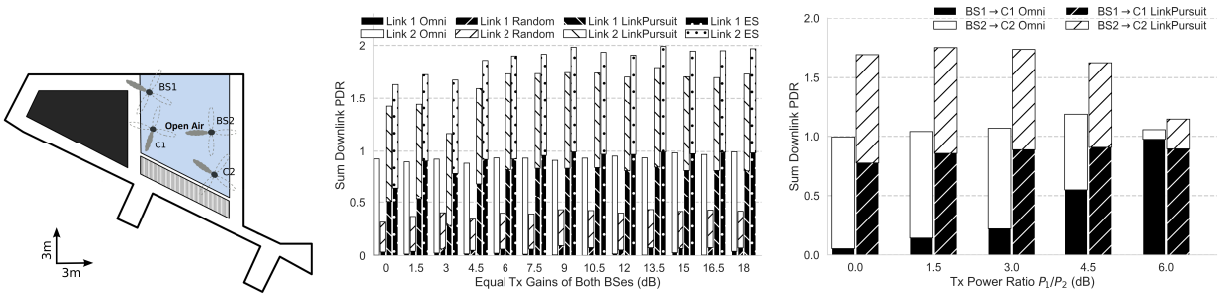
We set up a single BS-client link in a typical indoor office environment to verify LinkPursuit's ability to estimate and adapt to changing reward conditions. The BS uses a reconfigurable antenna, while the client operates with an omnidirectional dipole antenna. The real-time network operations are periodically frozen after 200 TDMA frames (6600 downlink packets) to inspect the pursuit statistics on the antenna adaptation process and induce artificial environmental changes, such as disabling (grounding control pin to reduce antenna gain) the currently perceived "optimal" Tx antenna mode. The verification procedure starts with all Tx directional modes active and then selectively disables among the remaining modes the current best Tx mode with the highest reward estimate after each 200-frame round.

Figure 5 shows the *reward estimates* and *selection counts* of the four possible directional Tx antenna modes as perceived and selected by LinkPursuit at the end of each round. Since we used an omnidirectional antenna at the receiving client, we aggregate the performance metrics of each Tx mode across all Rx antenna modes and present them. We observe that during each measurement round, the number of times a Tx antenna mode is selected consistently tracks its reward estimates. The current highest-reward Tx mode is selected the majority of the time, and a small fraction of time is necessarily spent exploring the remaining modes for adaptation. Once a mode becomes suboptimal, its reward is correctly updated within a single 200-frame round.

B. Impact of Directionality on Spatial Reuse

We quantify LinkPursuit's performance in an interference-limited environment with two concurrent BS-client links. We set up two BS-client links within the same vicinity indoor, under the measurement topology shown in Fig. 6a. All network nodes are equipped with reconfigurable antennas. We conduct transmission rounds of 100 TDMA frames, with 33 downlink and 1 uplink data slots per frame, and measure the downlink PDR and MAC goodputs MAC at each client. For each experiment, we compare LinkPursuit's performance to those of omnidirectional, random selection, and exhaustive search (ES) schemes. In the ES scheme, we sweep through all available directional antenna state combinations (16 for each link, and 256 for the two-link network) to determine *a posteriori* the optimal configuration which achieves the highest downlink sum rate.

In the first experiment, we sequentially increase the transmit powers of both BSs to generate stronger cross-link interference and observe the MAC layer goodputs. Figure 6b shows the downlink PDRs of the different transmission schemes across the range of Tx powers. In the Omni scenario, due to severe cross-link interference, only Link 2 can sustain a usable PDR. Client C1 is completely dominated by the interfering signals from BS2 and fails to deliver any MAC goodput. In contrast, the sum rate (total MAC goodput) of LinkPursuit consistently exceeds that of Omni by 74% on an average. Most of this sum rate improvement stems from the PDR



(a) Measurement setup **(b)** IF reduction at different Tx power levels **(c)** Downlink sum rates at different SINRs

Fig. 6: Two-link interference measurement results of the LinkPursuit protocol

increase of Link 1 - the weaker link in the Omni case. Furthermore, this spatial reuse gain of LinkPursuit persists across the range of transmission powers. We also note that LinkPursuit consistently delivers close to 90% of the sum rate achieved by the *a posteriori* search ES scheme. The remaining performance gap is within reason for our implementation of LinkPursuit adopts a 90/10 exploitation-exploration ratio.

To study LinkPursuit's performance under asymmetrical interference conditions, we keep the Tx power of BS2 constant at 15 dBm and gradually increase BS1's Tx power to generate fluctuating SINRs at both clients. Figure 6c depicts the downlink sum rate achieved under these conditions. In the Omni case, the throughput of a link is highly dependent on its Tx power ratio to the interference source. Due to its uniform proximity to both BSs, client C1 experiences stronger interference than C2 and yields much lower SINRs and throughput. This topology-dependent SINR loss can be compensated by increasing BS1's Tx power with respect to BS2, but at the price of a corresponding decrease in the second link's performance. In contrast, LinkPursuit delivers well-balanced, close to optimal link throughputs across the range of power ratios. This capability greatly simplifies the task of interference management and ensures reliable quality of service for users.

V. Conclusion and Future Work

We have presented the design, implementation, and evaluation of LinkPursuit, a novel learning protocol for distributed antenna state selection in directional small-cell networks. LinkPursuit incurs low overhead and adapts quickly to environmental changes through probabilistic selection at each time step. Our experimental results confirm that coordinated directional transmission provides significant advantages in terms of mitigating co-channel interference over omnidirectional transmission. However, LinkPursuit is not without limitations. The system optimizes link throughput greedily on a per-link basis, so it often disregards potential sum rate improvement from network cooperation. Future work can investigate automated parameter tuning of pursuit operations in terms of adaptation rate α and learning rate β, which will enable adaptive exploration-exploitation ratios and can further improve performance. Effort is also warranted to consider the benefits of network cooperation in LinkPursuit.

Acknowledgments

This work was supported by NSF under Grant No. 1457306 and Tekes under Grant Dnro 2336/31/2014.

References

[1] Y. Niu, Y. Li, D. Jin, L. Su, and A. V. Vasilakos, "A survey of millimeter wave communications (mmWave) for 5G: opportunities and challenges," *Wireless Networks*, vol. 21, no. 8, pp. 2657–2676, 2015.

[2] H.-N. Dai, K.-W. Ng, M. Li, and M.-Y. Wu, "An Overview of Using Directional Anatennas in Wireless Networks," *International Journal of Communication Systems*, vol. 26, no. 4, pp. 412–448, 2013.

[3] X. Liu, A. Sheth, M. Kaminsky, K. Papagiannaki, S. Seshan, and P. Steenkiste, "DIRC: Increasing Indoor Wireless Capacity using Directional Antennas," in *Proc. ACM SIGCOMM*, 2009.

[4] A. Amiri Sani, L. Zhong, and A. Sabharwal, "Directional antenna diversity for mobile devices : Characterizations and Solutions," in *Proc. ACM MobiCom*, 2010.

[5] Y. Gai, B. Krishnamachari, and R. Jain, "Learning multiuser channel allocations in cognitive radio networks: a combinatorial multi-armed bandit formulation," in *Proc. IEEE DySPAN*, 2010.

[6] A. Mukherjee and A. Hottinen, "Learning Algorithms for Energy-Efficient MIMO Antenna Subset Selection: Multi-Armed Bandit Framework," in *Proc. EUSIPCO*, 2012.

[7] N. Gulati and K. R. Dandekar, "Learning state selection for reconfigurable antennas: A multi-armed bandit approach," *IEEE Transactions on Antennas and Propagation*, vol. 62, no. 3, pp. 1027–1038, 2014.

[8] D. Thierens, "An adaptive pursuit strategy for allocating operator probabilities," in *Proc. ACM GECCO*, 2005.

[9] *WARP Project*, http://warpproject.org.

[10] D. H. Nguyen, A. Paatelma, H. Saarnisaari, N. Kandasamy, and K. R. Dandekar, "Enabling Synchronous Directional Channel Access on SDRs for Spectrum Sharing Applications," in *Proc. ACM WiNTECH*, 2016.

[11] P. Auer, N. Cesa-Bianchi, and P. Fischer, "Finite-time analysis of the multiarmed bandit problem," *Machine Learning*, vol. 47, no. 2-3, pp. 235–256, 2002.

[12] D. Patron and K. R. Dandekar, "Planar reconfigurable antenna with integrated switching control circuitry," in *Proc. EuCAP*, 2014.